For Macintosh Programmers

MacTech

MAGAZINE

l. 12, No. 7 • July 1996

MACTECH
7-96 JUL
MMM 6 9607
10.50
MAG7488700

n This
ssue:
ontrols!

INSIDE:

PROGRAMMING TOOLS
Symantec C++ v8 Release 5

WEBTECH™
CGIs in Lisp

BOOK REVIEW
Code Fragments Explained

JAVATECH™
Getting the Jump on JavaScript

PROGRAMMING TOOLS
Tools Plus 3.0

GETTING STARTED
A Java Popup Menu URL Launcher

SYMANTEC TOP 10

PROGRAMMER'S CHALLENGE
Connect IV

UNIFORM RESOURCE LOCATORS

FROM THE FACTORY FLOOR
A Little CodeWarrior History

CRABB'S APPLE
Too Many Cups of Java

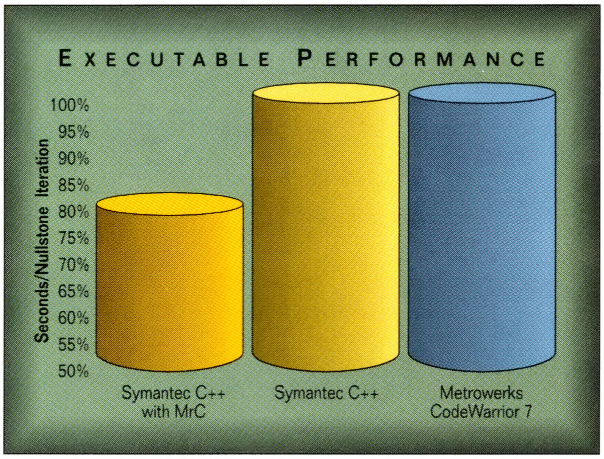
AND MORE!

$5.85 US
$6.95 Canada
ISSN 1067-8360
Printed in U.S.A.

WHEN MRC GOES TO WORK, SYMANTEC C++ APPLICATIONS GET 22% FASTER.

Now, through a joint development agreement, Symantec and Apple™ Computer let you produce the fastest Power Mac code.

Symantec C++ for Power Macintosh™ now comes with MrC– Apple's new optimizing compiler.



*Code compiled using MrC runs an average of 22% faster than code compiled with the standard Symantec C++ compiler or Metrowerks Code Warrior 7™.**

Industry-standard Nullstone tests show that applications compiled with MrC run an average of 22% faster.

DEVELOP FASTER APPLICATIONS FASTER.

Not only can you develop the fastest Power Mac applications, you can write them fast, too. New AppleScript support lets you automate repetitive tasks. While the new linker provides fast turnaround for incremental builds.

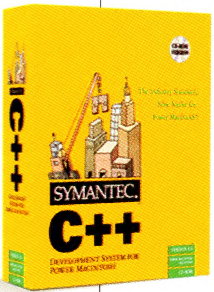
With the fully integrated class browser, you'll quickly navigate your C++ class library. And support for templates and multiple inheritance further boosts your productivity.

ORGANIZE AND MANAGE PROJECTS EASILY.

The new Project Manager lets you organize and manage nested projects. You can display hierarchical groups as folders within a project window for easier organization.

The editor gives you unlimited split panes and full text formatting– font, size and color. Organizing and navigating a project has never been easier.

In addition, Symantec C++'s multi-threaded environment gives you the ability to edit and write code while you compile. And our visual architect lets you quickly draw the interface. Corresponding code is then generated automatically.



MRC COMPILER
produces the fastest Power Mac applications

APPLESCRIPT
automates the build process

NEW INCREMENTAL LINKER
provides fast incremental builds

TEMPLATE AND MULTIPLE INHERITANCE SUPPORT
increases productivity

NESTED PROJECTS AND FOLDERS
let you organize and navigate projects

MULTI-THREADED ENVIRONMENT
lets you edit and write code while compiling

VISUAL ARCHITECT
builds your interface visually

All in all, Symantec C++ is a great way to develop the fastest applications for Power Mac.

THREE CDS FOR THE PRICE OF ONE.

When you register as a Symantec C++ owner, you'll be enrolled in the Symantec C++ Subscription for Macintosh program. Subscribers will automatically receive two free product updates (on CD ROM) so you'll always have the latest features and tools.

Learn more about Symantec C++ on the Internet at www.symantec.com
Or call 1-800-628-4777, Extension 9H25 for more information.

SYMANTEC.®

Offer valid in U.S.A. only. *Industry standard Nullstone Tests run 9/29/95 on Power Mac 6100/60. For more information in Canada, call 1-800-365-8541. In Australia, call 2-879-6577. In Europe, call 31-71-353111.

Symantec is a registered trademark of Symantec Corporation. All other trademarks are the property of their respective holders. All rights reserved. © 1995 Symantec Corporation.

Eddy Award Winner** for **Best New Developer Tool
– MacUser Editors Choice Awards, 1993

"A distinct improvement over ResEdit."
– MacTech / MacTutor

"Resorcerer's data template system is amazing!"
– Bill Goodman, author of Compact Pro

"Nuke ResEdit! Resorcerer is mission-critical for us."
– Dave Winer, Userland Frontier

"The color pixel editors are wonderful! A work of art!"
– Dave Winzler, author of Microseeds Redux

"Every Macintosh developer should own a copy of Resorcerer."
– Leonard Rosenthol, Aladdin Systems

"Resorcerer will pay for itself many times over in saved time and effort."
– MacUser review

"The template that disassembles 'PICT's is awesome!"
– Bill Steinberg, author of Pyro! and PBTools

"Resorcerer proved indispensible in its own creation!"
– Doug McKenna, author of Resorcerer

"…a wealth of time-saving tools."
MacUser Review, Dec. 1992

RESORCERER®

The Resource Editor for the Macintosh Wizard

Version 1.2.4

ORDERING INFO

Needs: ≥Mac Plus, ≥ Sys 4.2, 1MB
Likes: ≥Mac Plus, ≥ Sys 7.0, 2MB
32-bit clean, AU/X compatible

Price: $256 (decimal)
(Educational, quantity, or
other discounts available)

Includes: 500 page manual
60-day Money-Back Guarantee
Domestic UPS ground shipping

Payment: Check, PO's, or Visa/MC

Extras (call us):
COD, FedEx, UPS Blue/Red,
International Shipping

Downloadable Demos/Updaters:
AppleLink: Software Sampler
AOL: Software Libs/Development
CompuServe: MACDEV/Tools
or call us.

New 1.2 Features:

- **New 'cicn', 'ppat', 'crsr', 'acur', 'pltt', 'clut' editors**
- **Powerful icon family editing (all 9 icon types)**
- **Color pixel anti-aliasing, dithering, and lots more**
- **Complete 'PICT' disassembly and reassembly**
- **Resource sorting; ROM resource browsing**
- **120 template field parsing types now supported**
- **New insertion & deletion template field types**
- **Text-only 'PICT' resources**
- **Lots of improvements throughout**

- Easier, faster, more Mac-like, and more productive than ResEdit
- Safer memory-based, not disk-file-based, design and operation
- All file information and common commands in one easy-to-use window
- Compares resource files, and even **edits your data forks** as well
- Visible, accumulating, editable scrap
- Searches and opens/marks/selects resources by text content
- Makes global resource ID or type changes easily and safely
- Builds resource files from simple Rez-like scripts
- Most editors DeRez directly to the clipboard
- All graphic editors support screen-copying or partial screen-copying
- Hot-linking Value Converter for editing 32 bits in a dozen formats
- Its own 32-bit List Mgr can open and edit very large data structures
- Templates can pre- and post-process any arbitrary data structure
- Includes nearly 200 templates for common system resources
- TMPLs for Installer, MacApp, QT, Help, AppleEvent, OCE, GX, etc.
- Full integrated support for editing color dialogs and menus
- Try out balloons, 'ictb's, lists and popups, even create C source code
- Integrated single-window Hex/Code Editor, with patching, searching
- Editors for cursors, versions, pictures, bundles, and lots more
- Well-designed, helpful developer tools being added all the time
- Relied on by thousands of Macintosh developers around the world

MATHEMÆSTHETICS, INC.
P.O. Box 298 • Boulder • CO • 80306-0298 • USA
Phone: (303) 440-0707 • Fax: (303) 440-0504
AppleLink/AmericaOnline: RESORCERER • Internet: resorcerer@aol.com

For Macintosh
Programmers & Developers

MacTech™
MAGAZINE

A publication of XPLAIN CORPORATION

How To Communicate With Us

In this electronic age, the art of communication has become both easier and more complicated. Is it any surprise that we prefer **e-mail?** If you have any questions, feel free to call us at 805/494-9797 or fax us at 805/494-9798.

DEPARTMENTS	Internet	CompuServe	AppleLink
Orders, Circulation, & Customer Service	cust_service@mactech.com	71333,1063	–
Press Releases	press_releases@mactech.com	–	–
Ad Sales	ad_sales@mactech.com	–	–
Editorial	editorial@mactech.com	71333,1065	–
Programmer's Challenge	prog_challenge@mactech.com	–	–
Online Support	online@mactech.com	–	–
Accounting	accounting@mactech.com	–	–
Marketing	marketing@mactech.com	–	–
General	info@mactech.com	71333,1064	MacTechMag
Online support area	http://www.mactech.com		

MACTECH MAGAZINE

MacTech Magazine is grateful to the following individuals who contribute on a regular basis. We encourage others to share the technology.

We are dedicated to the distribution of useful programming information without regard to Apple's developer status.

*For information on submitting articles, ask us for our **writer's kit** which includes the terms and conditions upon which we publish articles.*

Editors

Publisher/Editor-in-Chief • Neil Ticktin
Editor • Matt Neuburg
Editorial Assistant • John Kawakami
Editor Emeritus • Scott T Boyd
Editor-at-Large • Eric Gundrum

Contributing and Technical Editors

Copland • Steve Kiene, Mindvision
Internet • Jon Wiederspan
MagicCap/TeleScript • Richard Clark, General Magic
OpenDoc • Tantek Çelik
Performance Programming • Jim Gochee
Tips & Tidbits • Steve Sisak
MacDev-1™ • Rich Siegel

Regular Columnists

Getting Started • Dave Mark
Programmer's Challenge • Bob Boonstra
Insider's View • Jordan Mattson
Symantec Top 10 • Symantec Technical Support
Crabb's Apple • Don Crabb
URLs • Jim Straus

XPLAIN CORPORATION

Chief Executive Officer • Neil Ticktin
Chief Operating Officer • Andrea J. Sniderman
Controller • Brian Shin
Advertising Executive • Ruth Subrin
Art Director • Judith Chaplin
Customer Service • Al Estrada
Vendor Relations • Duk Im
Administrative Assistant • Susan Pomrantz
Network Administrator • Donal Corcoran
Software Engineer • Don Bresee

Board of Advisors • Steven Geller, Jordan Mattson, Blake Park, and Alan Carsrud

PRINTED WITH SOY INK™

This publication is printed on paper with recycled content.

All contents are Copyright © 1984-1996 by Xplain Corporation. All rights reserved. MacTech, MacTech Magazine, MacTech CD-ROM, MacTech Web, THINK Reference, Developer Depot, Sprocket, JavaTech, WebTech, MacDev-1, Developer Central, Virtual Developer Central and the MacTutorMan are trademarks of Xplain Corporation. Other trademarks and copyrights appearing in this printing or software remain the property of their respective holders.

MacTech Magazine (ISSN: 1067-8360 / USPS: 010-227) is published monthly by Xplain Corporation, 850-P Hampshire Road, Westlake Village, CA 91361-2800. Voice: 805/494-9797, FAX: 805/494-9798. Domestic subscription rates are $47.00 per year. Canadian subscriptions are $59.00 per year. All other international subscriptions are $97.00 per year. Domestic source code disk subscriptions are $77 per year. All international disk subscriptions are $97.00 a year. Please remit in U.S. funds only. Second Class postage is paid at Thousand Oaks, CA and at additional mailing office.

POSTMASTER: Send address changes to **MacTech Magazine**, P.O. Box 5200, Westlake Village, CA 91359-5200.

BULK RATE
U.S. POSTAGE
PAID
PERMIT NO. 135
MIDLAND, MI 48640
THIRD ENCLOSED

TABLE OF CONTENTS

July 1996 • Volume 12, No. 7

For Macintosh Programmers & Developers
MacTech™
M A G A Z I N E

FEATURE ARTICLES

PROGRAMMING TOOLS 15
Symantec C++ v8 Release 5
A lot more than its Caffeine is eye-opening — *By Rich Parker*

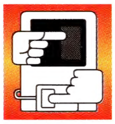
WEBTECH™ 25
CGIs in Lisp
A Web dictionary server — *By Mathieu Lafourcade and Gilles Sérasset*

BOOK REVIEW 46
Code Fragments Explained
A shared library of procedure pointers for the resourceful programmer — *By Mike Scanlin*

JAVATECH™ 48
Getting the Jump on JavaScript
Start learning to add life to your Web pages — *By Kevin M. Savetz*

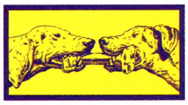
PROGRAMMING TOOLS 52
Tools Plus 3.0
Paradise for "control" freaks — *By Rich Parker*

REGULAR COLUMNS



GETTING STARTED 4
A Java Popup Menu URL Launcher — *By Dave Mark*

SYMANTEC TOP 10 33
— *By Craig Conner*

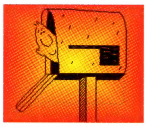


PROGRAMMER'S CHALLENGE 36
Connect IV — *By Bob Boonstra*

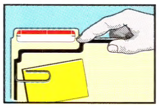
UNIFORM RESOURCE LOCATORS 51
— *By Jim Straus*

FROM THE FACTORY FLOOR 61
A Little CodeWarrior History — *By Dave Mark*

CRABB'S APPLE 66
Too Many Cups of Java — *By Don Crabb*

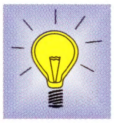


DIALOG BOX
68



THE CLASSIFIEDS
71

NEWSBITS
72



TIPS & TIDBITS
76

ADVERTISER & PRODUCT INDEX
78

GETTING STARTED

By Dave Mark

A Java Popup Menu URL Launcher

Since last month's column, I've been consumed with Java. I've been reading every Java book I could get my hands on, including the galleys for *Learn Java on the Macintosh* (Addison-Wesley), *Java in a Nutshell* (O'Reilly & Associates), *Active Java* (Addison-Wesley), *Java Essentials for C and C++ Programmers* (Addison-Wesley), and *Teach Yourself Java for Macintosh In 21 Days* (Hayden). There are lots of Java books out there, some of them much better than others.

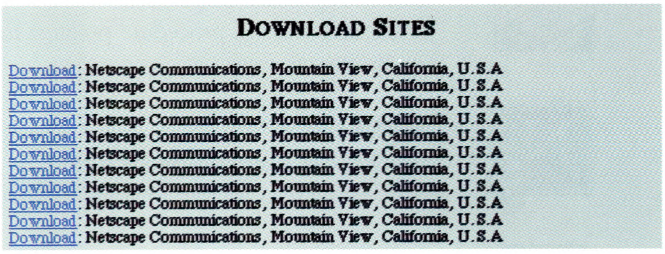
Beware of Java books written to pre-release versions of the Java SDK from Sun. Much has changed between the pre-release and release versions and, as a slew of messages on the net will attest, those changes will frustrate you as you attempt to get the book sample code to work.

The applets from *Learn Java on the Macintosh* and *Java in a Nutshell* have been translated into Metrowerks' Java project format. You can download the applets from Metrowerks' Web site (www.metrowerks.com).

THIS MONTH'S APPLETS

One important use of Java is to extend the functionality of a Web site, beyond the capabilities offered by HTML. If you've ever been to Netscape's Web site, you've probably encountered the long list of download sites that allow you to download the latest version of a Netscape product (Figure 1).

If you've got a long list of URLs to put on your Web site, you can, of course, represent them just like Netscape did, as a long series of links. The downside to this approach is that the links can take up a fair amount of screen space. An alternative is to use a form CGI and represent the list as a popup menu of URLs.



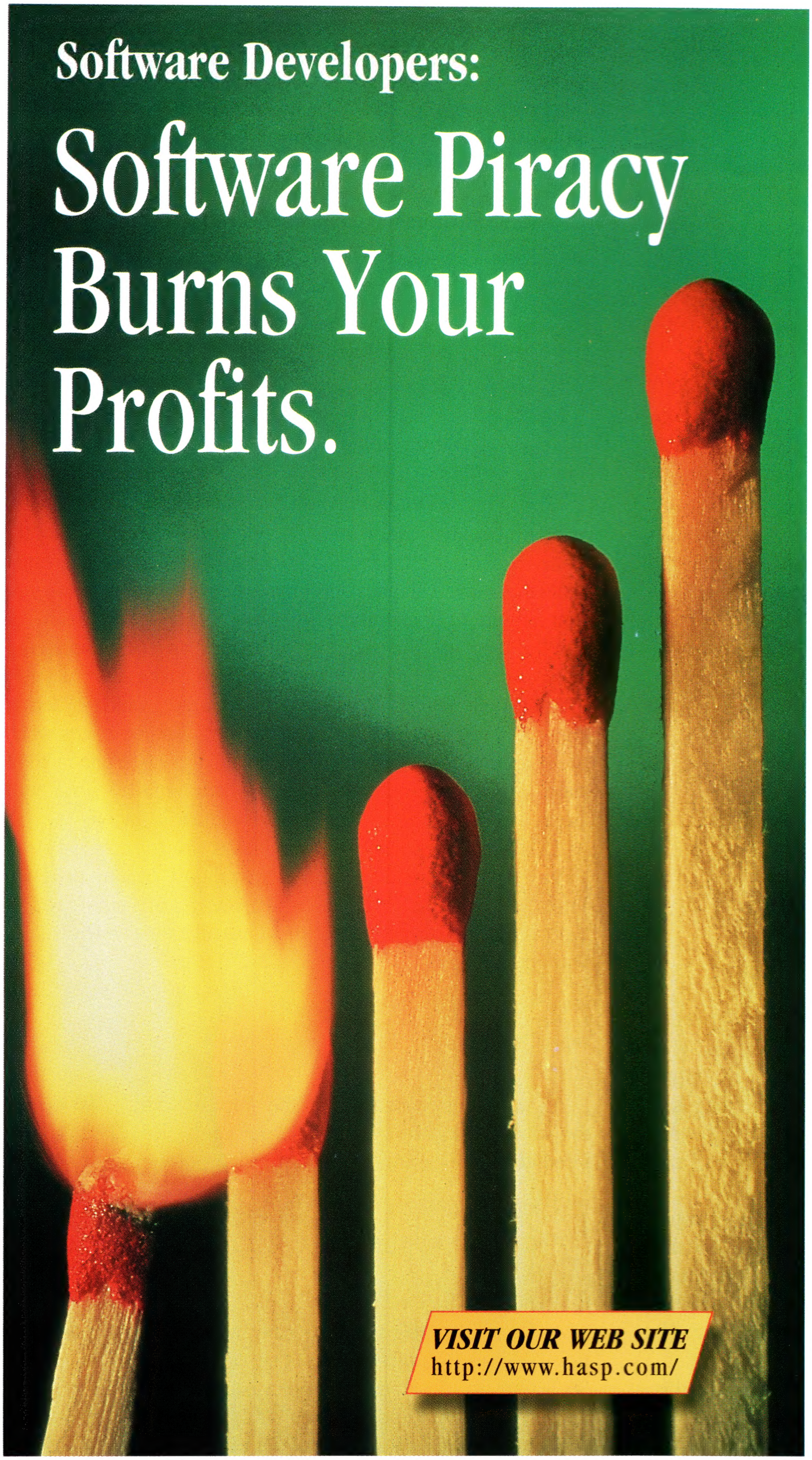
DOWNLOAD SITES

Download: Netscape Communications, Mountain View, California, U.S.A
Download: Netscape Communications, Mountain View, California, U.S.A
Download: Netscape Communications, Mountain View, California, U.S.A
Download: Netscape Communications, Mountain View, California, U.S.A
Download: Netscape Communications, Mountain View, California, U.S.A
Download: Netscape Communications, Mountain View, California, U.S.A
Download: Netscape Communications, Mountain View, California, U.S.A
Download: Netscape Communications, Mountain View, California, U.S.A
Download: Netscape Communications, Mountain View, California, U.S.A
Download: Netscape Communications, Mountain View, California, U.S.A
Download: Netscape Communications, Mountain View, California, U.S.A

Figure 1. The long list of sites to download the latest version of Netscape Navigator

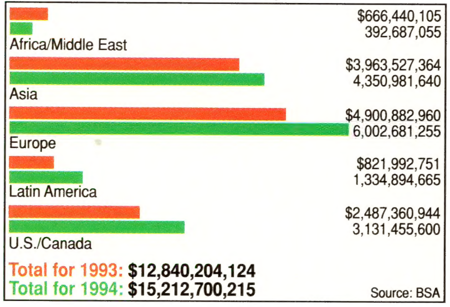
Solving the problem using a CGI is fine, but I wanted to implement this "popup" solution as a Java applet instead. To do this, we need to take advantage of the existing Java applet model, as embodied in the `java.applet.Applet` class. To make this a little easier to understand, I developed the applet in two stages. Here's the first version:

```
import java.applet.*;
import java.awt.*;
import java.net.*;

public class popup extends Applet
{
  private Choice  urlChoices;
  int          numURLs;

  public voidinit()
  {
    this.setBackground( Color.yellow );//java.awt.Color
    this.setForeground( Color.red );//java.awt.Color

    urlChoices = new Choice();

    urlChoices.addItem( "www.metrowerks.com" );
    urlChoices.addItem( "www.netscape.com" );
    urlChoices.addItem( "www.mactech.com" );

    urlChoices.setForeground( Color.green );
    urlChoices.setBackground( Color.blue );

    this.add( new Label( "Select URL: " ) );
    this.add( urlChoices );
  }

  public boolean action( Event e, Object obj )
  {
```



Each year, the illegal use of software consumes nearly 50% of your potential revenues. With the flames of piracy eating away at your profits, can you afford not to protect your software?

Software Obtained Illegally, by region, 1993 vs. 1994



MacHASP® is widely acclaimed as the world's most advanced software protection solution for Macintosh computers. Since 1984, thousands of leading Mac and PC developers have used over one million MacHASP and HASP keys to protect billions of dollars worth of software. Why? Because MacHASP's security, reliability, and ease-of-use led them to a simple conclusion: MacHASP is the most effective software protection system available.

Today, more developers are choosing MacHASP than any other software protection method. To learn why, and to see how easily you can increase your revenues, call now to order your MacHASP Developer's Kit.

1-800-223-4277

ALADDIN

The Professional's Choice

North America	**Aladdin Software Security Inc.** Tel: (800) 223 4277, 212-564 5678 Fax: 212-564 3377 E-mail: sales@hasp.com WWW: http://www.hasp.com/
Intl Office	**Aladdin Knowledge Systems Ltd.** Tel: 972-3-537 5795, Fax: 972-3-537 5796 E-mail: aladdin@aladdin.co.il
United Kingdom	**Aladdin Knowledge Systems UK Ltd.** Tel: 01753-622266, Fax: 01753-622262
France	**Aladdin France SA** Tel: 1 40 85 98 85, Fax: 1 41 21 90 56

© Aladdin Knowledge Systems Ltd. 1985-1995 *(6.95)* HASP® is a registered trademark of Aladdin Knowledge Systems Ltd. All other product names are trademarks of their respective manufacrurers. Mac & the Mac OS logo are trademarks of Apple Computer, Inc., used under license

member of

It runs with NetWare
Developer tested only

Available and Ready for OS/2

MICROSOFT® WINDOWS™ COMPATIBLE

Mac™OS

■ **Aladdin Benelux** 08894 19777 ■ **Aladdin Japan** 0426 60 7191 ■ **Aladdin Russia** 095 9230588 ■ **Australia** Conlab 3 8985685 ■ **Czech** Atlas 2 766085
■ **Chile** Micrologica 2 222 1388 ■ **Denmark** Berendsen 39 577300 ■ **Egypt** Zeineldein 2 3604632 ■ **Finland** ID-Systems 0 870 3520 ■ **Germany** CSS 201 278804
■ **Greece** Unibrain 1 6856320 ■ **India** Solutions 11 2218254 ■ **Italy** Partner Data 2 26147380 ■ **Korea** Dae-A 2 848 4481 ■ **Mexico** SiSoft 5 5439770
■ **New Zealand** Training 4 5666014 ■ **Poland** Systherm 61 480273 ■ **Portugal** Futurmatica 1 4116269 ■ **Romania** Interactiv 64 153112
■ **South Africa** D Le Roux 11 886 4704 ■ **Spain** PC Hardware 3 4493193 ■ **Switzerland** Opag 61 7169222 ■ **Taiwan** Teco 2 555 9676 ■ **Turkey** Mikrobeta 312 467 7504

```
    URL    url;

    if ( e.target == urlChoices )
    {
      try
      {
        url = new URL( obj.toString() );
      }
      catch( MalformedURLException err )
      {
        // Could print error message here.
        return true;
      }
      this.getAppletContext().showDocument( url );
      return true;
    }
    else
      return super.action( e, obj );
  }
}
```

Since I'm using CodeWarrior, I started by creating a new project file called `popup.μ` using the Java applet stationery. Next, I created a new source code file named `popup.java` and added it to the project. I typed in my source code, saved it, then selected the `.java` file in the project window and compiled it. The compiler used my Java source code to produce a file named `popup.class` which contains the Java byte code that will be loaded by my HTML file.

Speaking of my HTML file, I used CodeWarrior to create a new file called `popup.html` and saved it in the same folder as my other project files. I *didn't* add the HTML file to the project. Here's the HTML:

```
<HTML>
<HEAD>
<TITLE>URL Launcher</TITLE>
</HEAD>
<BODY>
This applet creates a popup menu of URLs.
<HR>
<APPLET CODE="popup.class" WIDTH=300 HEIGHT=35>
</APPLET>
</BODY>
</HTML>
```

If you've never seen HTML before, it's pretty straightforward. Basically, your HTML file contains the text you want to appear on your Web page, with a bunch of "tags" sprinkled throughout that define how the text looks. A tag is a pair of angle brackets that enclose a tag name, along with optional tag attributes. Some tags work in pairs. An end tag always starts with a `'/'`.

For example, to define the title of a particular page, you'll start with the `<title>` tag, follow it with the title, then follow the text with the `</title>` tag. The `<hr>` tag places a horizontal rule on your page. Note that the `<hr>` tag doesn't require a corresponding `</hr>` tag, since there is no text that needs to be surrounded. By the way, HTML tags are case-insensitive.

The `<applet>` tag is the tag we're interested in in this column. The `CODE` attribute specifies the name of the class file we want launched. In this case, we want the Java byte codes in `popup.class` executed.

Drag the HTML file on top of your favorite Java applet runner. If you have a Java-capable browser (like Netscape Atlas), you can open the HTML file and see how your applet looks on an actual Web page. Alternatively, if you run the applet using an applet runner, you'll see the applet portion of the Web page only. Most applet runners only understand the applet tag and will ignore the rest of the HTML.

Figure 2 shows the popup applet as viewed in an early beta of Netscape Atlas. As long as your Internet connection is live, selecting a URL from the popup menu transfers you to that URL.

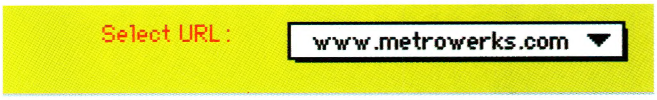


Figure 2. Our popup applet in action

Let's take a look at the source code...

THE POPUP.JAVA SOURCE CODE

As I already mentioned, this applet extends the `java.applet.Applet` class. Go into the API Documentation folder that came with your development environment and use your Web browser to open the file `java.applet.Applet.html`. As you'll see in the figure at the top of the page, `java.applet.Applet` is derived from `java.awt.Panel`, which is derived from `java.awt.Container`, which is derived from `java.awt.Component`. `java.awt.Component` is where all the action is. That's where you'll find all the interesting methods used by most applets. Spend a little quality time with `java.awt.Component.html`. It will definitely pay off.

The first few lines make the classes in `java.applet.*`, `java.awt.*`, and `java.net.*` available to our program. This allows us to refer to `Applet` instead of `java.applet.Applet` and `URL` instead of `java.net.URL`. You get the idea.

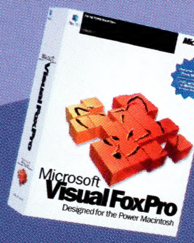
```
import java.applet.*;
import java.awt.*;
import java.net.*;
```

Our popup class extends the `Applet` class. We declare the variable `urlChoices` inside our class. `urlChoices` is a reference to a `Choice` (Java's term for a popup menu).

```
public class popup extends Applet
{
  private Choice  urlChoices;
```

The first method in our applet is the `init()` method. `init()` will be called automatically when our applet is started up. We'll use this method to set up our `Choice` popup menu. We'll start off by setting the applet background color to yellow and the foreground color to red. Note that `this` refers to the applet. If you take a look at Figure 2, you'll see that the background is yellow and that the popup menu's label text is red. I know yellow and red might not be a particularly good looking combination, but it does make the point, right?

Ever since Fred got the new Visual FoxPro for Power Macintosh, he catches the Frisbee every time.



And perhaps more importantly, he also gets his work done in record time. Thanks to Microsoft® Visual FoxPro™ 3.0 for Power Macintosh,® one might say that Fred's productivity is positively soaring.

After all, the powerful object technology of Visual FoxPro lets him create class libraries with inheritance and encapsulation and therefore reuse code quickly and easily. Fred has no trouble integrating data, either, whether it's located in applications, client/server databases, or local datastores. And because of Visual FoxPro's complete binary compatibility, he can develop in the familiar Mac™ environment and rest assured that all code, including Visual FoxPro objects, will run in Windows® 3.1 and Windows 95 operating systems with no modifications.

To find out more about the power of the Visual FoxPro 3.0 database management system, just point your mouse to http://www.microsoft.com/devonly/. Or give us a call at **(800) 621-7930, Dept. A334DS**, in the 50 United States.*

Microsoft®
WHERE DO YOU WANT TO GO TODAY?™

***In Canada call (800) 563-9048**; outside the U.S. contact your local subsidiary. ©1996 Microsoft Corporation. All rights reserved. Microsoft and Windows are registered trademarks and Visual FoxPro and *Where do you want to go today?* are trademarks of Microsoft Corporation. Power Macintosh is a registered trademark and Mac is a trademark of Apple Computer, Inc. Frisbee® is a trademark of Mattel, Inc. Used with permission.

```
  public voidinit()
  {
    this.setBackground( Color.yellow );//java.awt.Color
    this.setForeground( Color.red );//java.awt.Color
```

The next line uses `new` to create a new `Choice` object. Note that `urlChoices` is set to null until the new `Choice` is created.

```
    urlChoices = new Choice();
```

Next, the `Choice` classes' `addItem()` method is called to add three URLs to the project. The popup menu's foreground and background colors are set, too, but these calls don't seem to have any effect.

```
    urlChoices.addItem( "www.metrowerks.com" );
    urlChoices.addItem( "www.netscape.com" );
    urlChoices.addItem( "www.mactech.com" );

    urlChoices.setForeground( Color.green );
    urlChoices.setBackground( Color.blue );
```

Finally, a new `Label` object is created and the `Label` and `Choice` objects are added to the applet.

```
    this.add( new Label( "Select URL: " ) );
    this.add( urlChoices );
  }
```

The `action()` method is called whenever an event occurs that relates to the applet. We'll declare a `URL` reference to hold the URL we want to transfer to.

```
  public boolean action( Event e, Object obj )
  {
    URL   url;
```

The `Event` object `e` contains the event description we are responding to. Check out `java.awt.Event` to get a sense of Java event handling. `e.target` specifies the target object of the event. If the target is our popup menu, `urlChoices`, we'll handle the event. Otherwise, we'll pass the event on to our superclass and return whatever value `super.action()` returns.

If `urlChoices` was the target, we'll try creating a new `URL` object. Some Java objects require that you catch exceptions thrown by the object. To get a sense of this, go to the `java.net.URL.html` page. Note that all of the constructors throw the `MalformedURLException` error. You must catch this exception or your code won't compile. We ignore the error, but we could print an error message or, better yet, put up a dialog letting users know that the URL they selected isn't quite right.

If you throw an exception, you'll create a new `java.lang.exception` object (or some class that extends `throwable`). But when you catch an exception, the exception object is passed to your `catch` block, as `err` was in this case. Take a look at `java.lang.Throwable.html`, especially the `getMessage()` method.

```
    if ( e.target == urlChoices )
    {
      try
      {
```

Since the event target is our popup menu, the object will be the menu itself. By calling the method `toString()` (inherited from `Object`), we'll get a string with the popup menu's current setting. We'll then use that string to initialize the `URL`.

```
        url = new URL( obj.toString() );
      }
      catch( MalformedURLException err )
      {
        // Could print error message here.
        return true;
      }
```

Assuming the `URL` object was created okay, we'll call `getAppletContext()` to get our applet's context, then use that object to call `showDocument()`, which will tell our browser to jump to a different URL.

```
      this.getAppletContext().showDocument( url );
      return true;
    }
    else
      return super.action( e, obj );
  }
}
```

SAME APPLET, USING PARAMETERS

Here's a second version of the applet that gets its URLs from the HTML file. To see how this works, take a look at the second version of our HTML file:

```
<HTML>
<HEAD>
<TITLE>URL Launcher</TITLE>
</HEAD>
<BODY>
This applet takes a series of parameters and turns them into
a popup menu of URLs. The PARAM with the name numURLs tells
you how many URLs are included. PARAM "1" contains the 1st
URL, PARAM "2" contains the next URL, etc.
<HR>
<APPLET code="popup.class" width=300 height=35>
<PARAM name="numURLs" value="3">
<PARAM name="1" value="http://www.metrowerks.com">
<PARAM name="2" value="http://www.mactech.com">
<PARAM name="3" value="http://www.netscape.com">
</APPLET>
</BODY>
</HTML>
```

Notice that we've added a series of `<PARAM>` tags between the `<APPLET>` and `</APPLET>` tags. Each `PARAM` passes a named parameter to our applet. The `numURLs` parameter tells our applet how many URLs we are passing in. The `1` parameter contains the first URL, `2` contains the second URL, etc.

Here's the second version of our applet:

```
import java.applet.*;
import java.awt.*;
import java.net.*;

public class popup extends Applet
```

Announcing

MICROGUARD PLUS™

Why so many developers are switching to MicroGuard copy protection

MicroGuard is committed to uncompromising technological superiority

"Technology at its peak" is our commitment to you. That is why we have brought you **MicroGuard Plus**™. And, MicroGuard Plus is **100% backwards** compatible with MicroGuard. This means MicroGuard and MicroGuard Plus can be used interchangeably. Just as we promised!

MicroGuard offers you the most sophisticated network protection

Our network protection, **MicroGuard Net**™, is so superior, we had to hire an Apple network engineer to execute our specifications.

MicroGuard is the only key developed by Mac developers, and is the first and only 100% ADB savvy key

We have been developing Mac applications as a seed development house since 1984. We are not a PC protection company that has come to you with a Mac product. MicroGuard is fully ADB savvy and offers extended addressing. Unlike other protection devices, MicroGuard never clashes with other keys. Only MicroGuard offers this level of sophistication.

MicroGuard has now surpassed its own technological lead, actually improving on the best

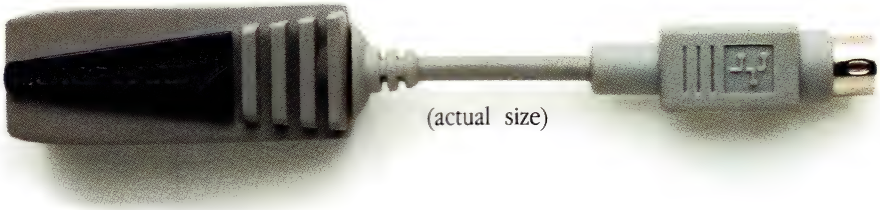
MicroGuard Plus is everything MicroGuard is, plus 40-bit encryption, two additional passwords, 64-Bit Array, 32-byte public area, 45% size reduction, enhanced counter and more. In addition, MicroGuard Plus offers two new utilities: **QuickGuard**™ and EasyGuard™ allow you to protect your applications without touching your source code. The only feature that is not plus is the price. :-)

MicroGuard is the best selling Macintosh key in the world

MicroGuard sells more Macintosh copy-protection keys than anyone else in the world!

MicroGuard delivers developer support within 24 hours

We will answer any inquiry you have within 24-hours. We also have a fully loaded AppleLink bulletin board which contains all our libraries, tech notes, Q&A and nearly everything you'll ever need!



(actual size)

For more information and to order a Developer's Kit or to receive a free CD ROM about MicroGuard, please contact us at:

MicroGuard USA: Tel: (303) 320-1628 • Fax: (303) 320-1599 • AppleLink: M.GUARD
International: Tel: (972) 3 558-2345 • Fax: (972) 3 558-2344 • AppleLink: MICROGUARD

Mac™OS

Pirate Busters

```
{
  private Choice  urlChoices;

  public voidinit()
  {
    int    numURLs = 0;

    this.setBackground( Color.yellow );//java.awt.Color
    this.setForeground( Color.red );//java.awt.Color

    urlChoices = new Choice();

    urlChoices.setForeground( Color.green );
    urlChoices.setBackground( Color.blue );

    String numURLsString = this.getParameter( "numURLs" );

    if ( numURLsString == null )
    {
      this.showStatus( "null or missing numURLs Parameter!"
);
    }
    else
    {
      try
      {
        numURLs = Integer.parseInt( numURLsString );
      }
      catch( NumberFormatException err )
      {
        numURLs = 0;
        this.showStatus( "Bad numURLs Parameter: " +
                         numURLsString );
      }

      String   urlString;

      for ( int i=1; i<=numURLs; i++ )
      {
        urlString = this.getParameter( Integer.toString( i )
);

        if ( urlString == null )
          this.showStatus( "Missing Parameter: " + i );
        else
          urlChoices.addItem( urlString );
      }
    }

    this.add( new Label( "Select URL: " ) );
    this.add( urlChoices );
  }

  public boolean action( Event e, Object obj )
  {
    URL    url;

    if ( e.target == urlChoices )
    {
      try
      {
        url = new URL( obj.toString() );
      }
      catch( MalformedURLException err )
      {
        // Could print error message here.
        return true;
      }
      this.getAppletContext().showDocument( url );
      return true;
    }
    else
      return super.action( e, obj );
  }
}
```

Basically, the difference here is in this chunk of code from `init()`:

```
    String numURLsString = this.getParameter( "numURLs" );

    if ( numURLsString == null )
    {
      this.showStatus( "null or missing numURLs Parameter!"
);
    }
    else
    {
      try
      {
        numURLs = Integer.parseInt( numURLsString );
      }
      catch( NumberFormatException err )
      {
        numURLs = 0;
        this.showStatus( "Bad numURLs Parameter: " +
                         numURLsString );
      }

      String   urlString;

      for ( int i=1; i<=numURLs; i++ )
      {
        urlString = this.getParameter( Integer.toString( i )
);

        if ( urlString == null )
          this.showStatus( "Missing Parameter: " + i );
        else
          urlChoices.addItem( urlString );
      }
    }

    this.add( new Label( "Select URL: " ) );
    this.add( urlChoices );
```

We start by calling `getParameter()` to retrieve the `numURLs` parameter. If the parameter is missing, we call `showStatus()` to display an error message in the browser's status message area. If the parameter was there, we'll try parsing an integer value from the string, placing the value in `numURLs`. Assuming we get a proper number, we'll step from 1 to `numURLs`.

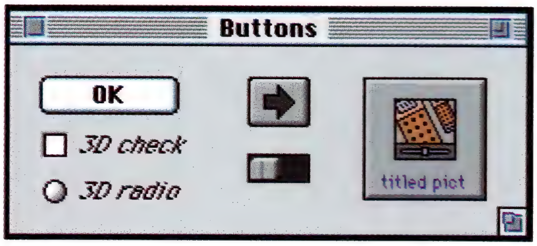
Inside the loop, we'll convert the current index into a string (using `Integer.toString()`) and pass that string into `getParameter()`. If that parameter exists, we'll add that parameter into the popup menu.

TILL NEXT MONTH...

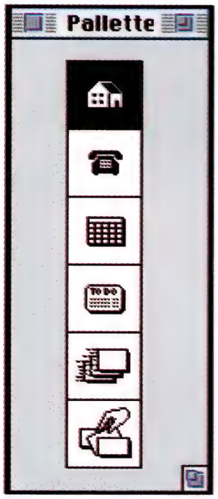
Want more Java? Want more PowerPlant? Want some TCL? What the heck *do* you want? Send me some email. Let me know. Otherwise, I'll just keep writing about Java until I get sick of it! <g> By the way, I hope that you will take a moment and visit www.zumacafe.com, the Web site for a brand new internet cafe in Santa Fe, New Mexico. Zuma's is the brainchild of Robin Williams, a brilliant writer and someone who has given a great deal to the Macintosh community. Check out the site and, if you get to Santa Fe, drop by Zuma's and say hello!

AppMaker

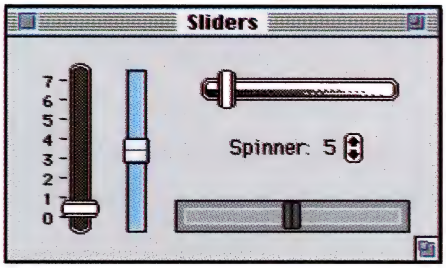
Your Assistant Programmer



AppMaker makes it fast and easy to build a user interface. Just point and click, then let AppMaker create resources and generate source code. AppMaker generates excellent source code for most popular languages and frameworks.



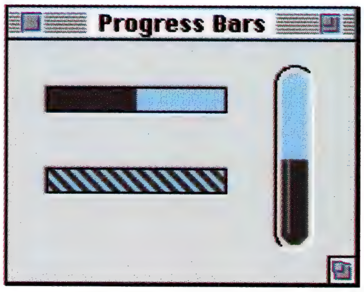
It's like having an expert assistant working for you. Use it as a prototyping and productivity tool. You concentrate on the unique features of your application instead of the standard parts. Novices read the AppMaker code to learn Macintosh programming techniques.



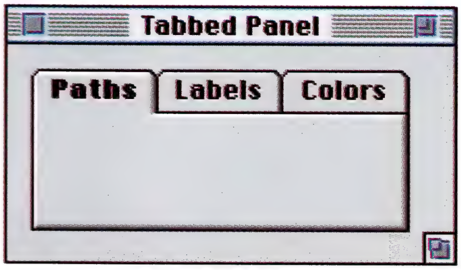
AppMaker now supports a wide variety of gadgets including 3D controls, picture buttons, sliders, progress bars, and tabbed panels. Users can easily customize and extend AppMaker to support other custom CDEFs.

The built-in simulator lets you try out all these new gadgets.

Interested in OpenDoc and the OpenDoc Development Framework (ODF)? AppMaker makes it easy to get started. Just design your interface then generate code to make an OpenDoc part editor.



AppMaker also supports CodeWarrior PowerPlant, procedural C, and PowerMac development. A single AppMaker document can now generate code and resources for a wide variety of languages, frameworks, and eventually, platforms.



Includes one-year automatic update service with at least three CDs per year.

Available from MacTech Mail Order Store for just $299 with a 60-day money-back guarantee.

Demo versions on AOL: keyword Bowers, on Compuserve in the MacDev forum, and at ftp://members.aol.com/bowersdev.

B•O•W•E•R•S
Development

P.O. Box 929, Grantham, NH 03753 • 603-863-0945
bowersdev@aol.com • 70731.3710@compuserve.com

Our competitors' habit of charging for each piece of their installer program is a bit puzzling.

- Create up to 128 Expert-Install "packages"
- Improved Developer Customization

The reason we call StuffIt InstallerMaker 3.0 the Complete Installation Solution is because we include **everything you need** to prepare a professional package of files for installation on your user's machines. An installer, updater, and an uninstaller are just three components included for the same price one of our fine competitors charges for just the installer. Using our installer reduces technical support calls due to end-user errors during installation, saves disks (if distributing on floppies) or download time (if distributing on-line). You will save more than the cost of the Installer license by significantly reducing distribution costs. And the puzzle will be solved. You'll have every piece in place.

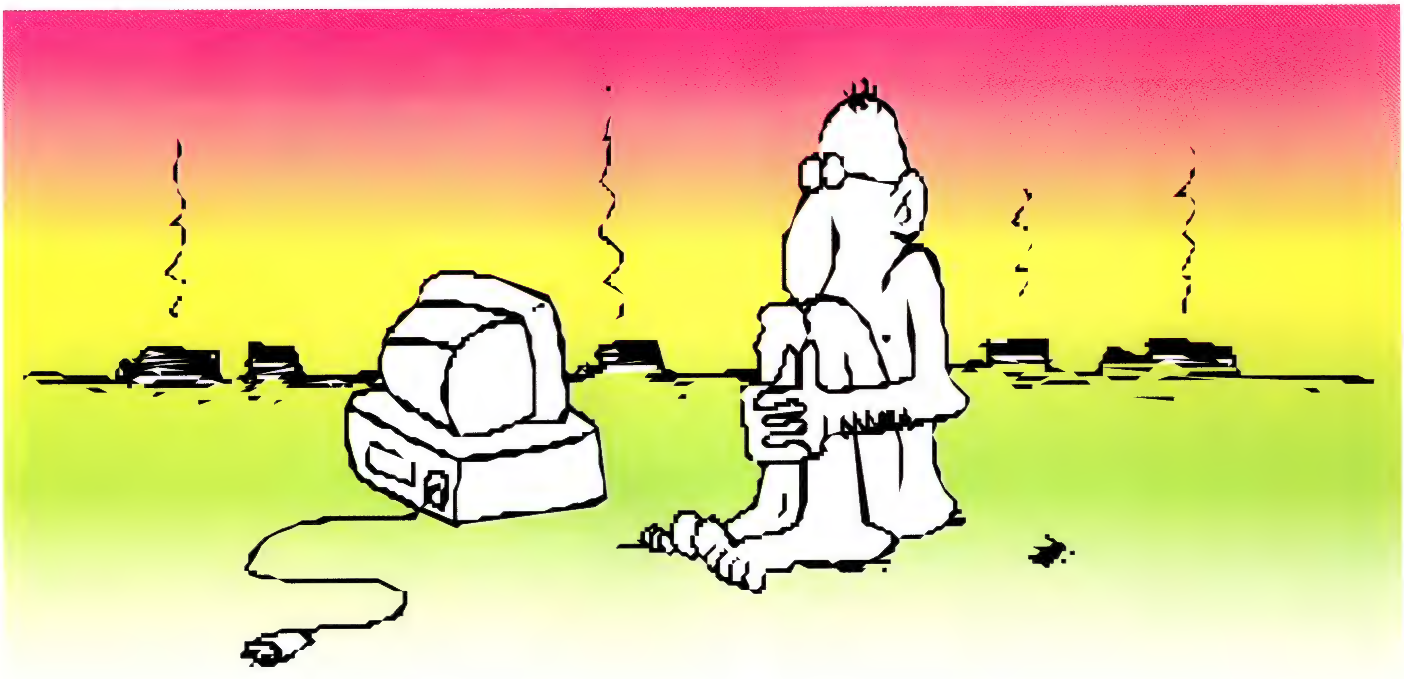
Download a **free,** fully-functional copy of StuffIt InstallerMaker 3.0 from www.aladdinsys.com, or call (408) 761-6200 and ask for Developer Sales.

StuffIt InstallerMaker 3.0

© 1996 Aladdin Systems, Inc. 165 Westridge Drive, Watsonville, CA 95076. Fax: (408) 761-6206. Internet: cust.service@aladdinsys.com. AOL, AppleLink: ALADDIN. StuffIt InstallerMaker is a trademark of Aladdin Systems, Inc. Other products are trademarks of their respective holders.

- Uninstall
- Built-in Resource Compression
- Built-In Updaters
- Custom Destinations
- Full Scripting and Recording
- Best Compression Available
- Resource Installation
- Move/Copy/ Rename any File

Aladdin Systems



Do I have to use a PC to port Windows applications to a Mac?

WILLOWS TWIN APIW for Macintosh

Windows API tool box extension for Macintosh developers

- Multi-platform implementation of the Windows API
- Works with Macintosh tools and compilers
- Native library optimized for PowerPC or 68K
- Work within the IDE of your choice

Call now: (408) 777-1820
Visit our web site: http://www.willows.com

Come and see us at MacWorld!

WILLOWS
SOFTWARE

Willows Software, Inc. 12950 Saratoga Avenue, Suite A, Saratoga, CA 95070
All trademarks are trademarks of their respective companies.

By Rich Parker

Symantec C++ v8 Release 5

A lot more than its Caffeine is eye-opening

ABOUT SYMANTEC'S NEW RELEASE

Symantec C++ version 8 release 4 was the beginning of Symantec's new subscription program for this product. I described it in considerable detail in my article, "New Symantec Development Tools" (*MacTech Magazine* 11.12 [December 1995] 43–50), and I'm not going to repeat all of that here.

Doubtless you're interested in what's new with Symantec C++ and the Symantec Project Manager (SPM), and there's lots to report on that front. The new SPM has a *much* smaller footprint. The previous version required about 12MB of RAM to run; the new version contained in this release has a preferred size of 6.5MB. That's a reduction of almost 50%! In addition, the new version requires less system temporary memory for precompiled headers, and that space is managed much more efficiently.

Another big step forward with the new release of the SPM is that you can develop both 68K and Power Mac applications from the *same* project file. This is a *huge* advantage for those of us who had to keep two project files and try to keep them both up to date.

There have been bug fixes and other changes to the Visual Architect and the TCL in this new version. I'm not aware of any new classes, but the existing classes have been made even more robust. One thing that users of Object I/O will appreciate is the inclusion of the "Using Object I/O" document, which goes into a quite a lot of detail about how this feature works and how it can be added easily to your application's classes. The document is in Acrobat format inside the "Documentation" folder, within the "EIC" folder of the "Symantec C++ for Power Mac" folder.

In addition to the "Latest Symantec Tools" folder's contents, there are quite a number of Apple software products included in the new release. There is a complete hard-disk-ready copy of MPW v3.4, including all of its compilers and tools, the latest version of MacApp (3.3), ToolServer, SourceServer, Apple debuggers, ResEdit, and a "SOM Objects for Mac OS" folder that contains the shared library, plus Apple's statement of direction in their development plans. Code, interfaces, and examples for MPW are also included, and there are interfaces and libraries for QuickDraw 3D and drivers and interfaces for the PCI bus. There are a lot of additional Apple components that I haven't mentioned.

BUT WAIT, THERE'S MORE…

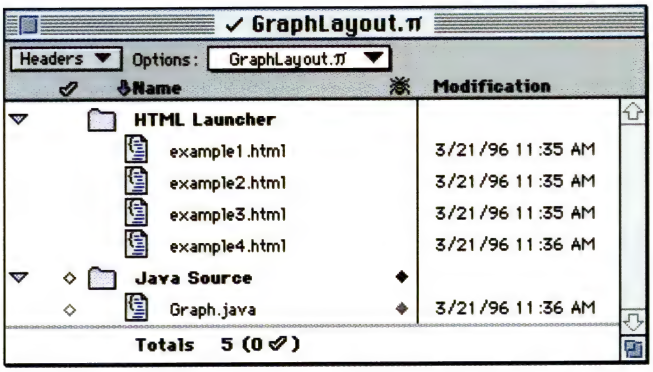
Release 5 includes Caffeine, a translator plug-in that supports the development of Java Applets with the SPM. You will need to use the Custom Install option to install Caffeine, which also installs Sun Microsystems' Java compiler and Applet Viewer code, as well as a ton of examples.

Caffeine is a prelude to the announcement of Café for the Macintosh, a stand-alone development environment for Java applets and applications. Symantec delivered Espresso for the Windows platform last December. This, like Caffeine, was a plug-in feature for their v7.21 C++ development environment. Symantec released the stand-alone version of Café for Windows late in March and expects to release Café for the Mac some time soon – perhaps it will have been released as you read this review. [*As indeed it was. While Rich, at our urging, was madly laboring to cover Caffeine in timely fashion, Café for Mac burst upon the scene, to be sent free to all Symantec C++ subscribers – thus truly superseding Caffeine. The rapidity of events is proving somewhat overwhelming. Nonetheless, we have not excised Rich's discussion, as it would have been wrong to rob our readers of its cogency and inherent interest. – man*]

For author information, see "Tools Plus 3.0", p. 52.

At this point, Caffeine relies on version 1.0b1 of Sun's Java Development Kit (JDK). That kit has some outstanding problems, not the least of which is that audio files aren't supported. There are also other bugs in Sun's software. I'm told that Sun intends to release the second beta version of their JDK sometime in April, and that the audio problem, among others, is solved.

Even though Sun's software is not quite up to par at this point, you can still develop applets that will run quite well with Java-equipped Internet browsers, or with the included Applet Viewer. To give you an idea of a Java applet, Figure 1 illustrates a typical SPM project file window for Java applet development.



✓	Name		Modification
	HTML Launcher		
	example1.html		3/21/96 11:35 AM
	example2.html		3/21/96 11:35 AM
	example3.html		3/21/96 11:35 AM
	example4.html		3/21/96 11:36 AM
◇	Java Source	◆	
◇	Graph.java	◆	3/21/96 11:36 AM
	Totals 5 (0 ✓)		

Figure 1. SPM project for Java development

As you can see from Figure 1, a Java project includes one or more HTML files and a Java source file. The various HTML files can be used to supply different sets of parameters to the Java applet. To compile the project, choose **Bring Up To Date** from the **Build** menu. This causes the SPM to invoke Sun's Javac compiler to compile the source file (in this case, `Graph.java`). When the compilation is complete, the compiled classes are stored inside the "Classes" folder of the "MacJDL 1.0b1" folder that was installed when you installed Caffeine. I compiled several of the examples and all of the compiled classes are just lumped together in that one folder. It would have been better if they were separated into individual folders, by project name. To execute the applet, you need either to move all of the classes and the HTML files into the place where your Internet browser expects to see them, or to leave them where they are and choose the **Run Applet (example1.html)** command from the **AppleScript** menu. This will cause the Applet Viewer application to be invoked with the "example1.html" file. If you wish to see the results of running any of the other HTML files, then you'll have to either duplicate and modify the script to refer to "example2.html," "example3.html," and so forth, or open the individual HTML files from within the Applet Viewer, once it has been invoked. The result of running the "example1.html" file by choosing the **Run Applet** script is shown in Figure 2.

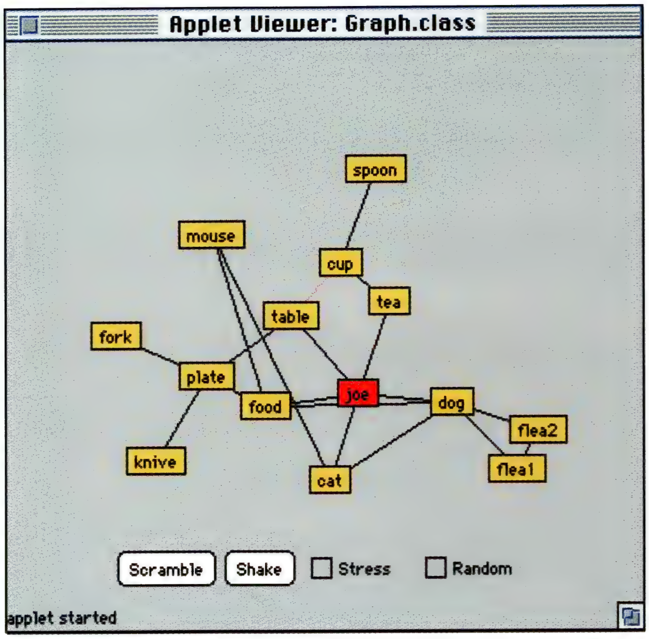
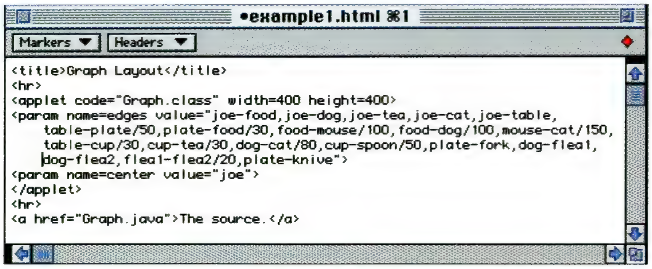


Figure 2. Graph layout applet in execution

The static picture in Figure 2 doesn't do the applet justice. The positions of the various nodes of the graph are set initially to random positions and then, during the applet's execution, the nodes eventually drift into a steady state, which is the point at which the figure was captured. If you click the **Scramble** button, then the node-locations are scrambled and the graph again drifts to a (perhaps different) steady state. The **Shake** button, when clicked, causes the nodes to be shaken up and down to reposition them, and then they gravitate to another steady state. The **Stress** and **Random** checkboxes cause the nodes to behave differently, although I'm not quite sure I understand the purpose of these settings. Nevertheless, the Graph Layout is only one of the many examples of dynamic Java applets that were written by Sun and included in Symantec's Caffeine release.

Figure 3 shows the contents of the "example1.html" file that was used to specify the parameters of the graph to the compiled applet. I arbitrarily chopped the `<param>` directive into multiple lines for purposes of showing the entire specification in the figure.



```
<title>Graph Layout</title>
<hr>
<applet code="Graph.class" width=400 height=400>
<param name=edges value="joe-food,joe-dog,joe-tea,joe-cat,joe-table,
    table-plate/50,plate-food/30,food-mouse/100,food-dog/100,mouse-cat/150,
    table-cup/30,cup-tea/30,dog-cat/80,cup-spoon/50,plate-fork,dog-flea1,
    dog-flea2,flea1-flea2/20,plate-knive">
<param name=center value="joe">
</applet>
<hr>
<a href="Graph.java">The source.</a>
```

Figure 3. Example1.html file's contents

neoAccess™
Cross-Platform Object Database Engine

Power Too Abundant to Meter

Powerful

NeoAccess and NeoShare are the most powerful object-oriented database engines available. They display electrifying performance—up to ten times that of competitors. Behind an elegant programming interface is a high performance query engine utilizing: extended binary trees and binary search algorithms tuned for short access times, dynamically combined, collapsed, and compressed indices, and object caching for lightning fast access to previously used objects.

No Runtime Fees

Get the power of NeoAccess and avoid the expense and administrative hassle of feeding the runtime fees meter. You pay one affordable price no matter how many copies of your application you sell or use.

Cross Platform

Others may promise cross-platform development tools—NeoLogic delivers. NeoAccess and NeoShare consists of C++ classes designed for use with popular compilers and application frameworks on Windows® (3.x, '95 and NT), Macintosh® (M68k and PPC), and Unix™ (all flavors) platforms. Full source code is available so both products can even be used with custom frameworks.

New in NeoShare Version 2.0

Cross-platform and internet-savvy NeoShare 2.0 is now shipping! NeoShare extends standard application frameworks including; Metrowerks' PowerPlant, Symantec's THINK Class Library and Apple's MacApp on the Macintosh, and Microsoft's Foundation Classes and Borland's OWL in Intel-based environments.

Internet-savvy NeoShare is Here!

neo•logic *Now!*

http://www.neologic.com
Download the Architectural Overview!

neoShare™
Client/Server Object Database Engine

Scalable

NeoShare extends the core features of NeoAccess to client/server applications. NeoShare includes the NeoAccess Toolkit and everything you need to create data intensive distributed applications. Its advanced client/server architecture provides shared access to objects by multiple inter- and intra-networked clients. Combine client and server functions into a single application or create separate client and server applications. With NeoShare you have complete flexibility in the design of your distributed systems.

Proven

Thousands of commercial and in-house developers have already found that NeoLogic's technology enabled them to build fast, lightweight and powerful applications in record time. That's why NeoAccess and NeoShare based applications are already operating on millions of computers. Tap into the power for your next development project!

neo•logic™

Powering Development of Object-Oriented Applications

NeoLogic Systems
neologic@neologic.com

1450 Fourth St., Suite 12
Berkeley, CA 94710

v. 510.524.5897
f. 510.524.4501

```
"DATA 'FKEY' (6) {
    $"514F A924 A873 3F3C"
    $"BF86 A98C 2057 2050"
    $"4868 0002 A8F6 4E75"
}; "
```

56 million people have developed a passion for Macintosh. You, however, may end up developing something more.

Macintosh users have always been passionate about their computers. In fact, 90% of them buy Mac® computers again when choosing a second system—and 39% actually call themselves "missionaries" for the brand.

Bottom line: When it comes to customer loyalty and satisfaction, no other brand even comes close. People who use Macintosh, quite simply, love it.

Which explains why, in 1995, we sold more Apple® Macintosh® computers to more people than ever before.

Welcome to Apple Developer Relations

The reason so many people love what they can get out of Macintosh is that developers like you put a lot into it.

And what you can put into it are ideas—big ideas that become distinctly superior customer solutions.

To help you turn your big ideas into reality, Apple provides an extensive library of resources.

Our World Wide Web site includes technical strategies and documentation, software developer kits, on-line training, and marketing information.

For those who want more, we also offer a wide range of development tools, classroom training, and support programs.

And once your product is complete, we can help you reach millions of customers.

Passionate customers.

So whether you're a newcomer to Macintosh or an experienced Mac developer, there's a good chance we'll have exactly what you're looking for.

Your Success Is Our Commitment

It's important to us that you succeed, because we know your ideas can make a difference.

That's why our commitment to developers is so strong, and why we offer as much support as we do.

So take a few moments to visit our Web site today. Or call for a free copy of the Apple Developer Catalog.

And see for yourself how much more there is to Macintosh.

Contact us at 1-800-282-2732 or, http://dev.info.apple.com.

Apple Developer Relations

Outside the US, call: Australia +61 2.452.8000 Belgium, Luxembourg +32 2.4744290 Brazil 011.5511.886.8000 Asia 852 2506.8888 France 05.90.72.26 Germany, Austria +49 89.99640.534 Italy+39 2.273.26.1 Japan +81 3.5717.7520 Latin America+1 305.569.2759 The Netherlands +31 30.6593890 Nordic Region+46 8.703.30.99 Spain +34 1.663.17.80 Switzerland+41 1.877.9119 United Kingdom, Ireland 0800.50.50.94, +44 131.458.6736 Canada+1 800.637.0029.
If you are located in Africa, Central Europe, Eastern Europe, the Mediterranean, or the Middle East, please call your local Apple representative, or look on our web site for further information.

© 1996 Apple Computer, Inc. All rights reserved. Apple, the Apple logo, and Macintosh are trademarks of Apple Computer, Inc. registered in the U.S. and other countries. Mac is a trademark of Apple Computer, Inc.

JAVA

MACINTOSH

ROASTER

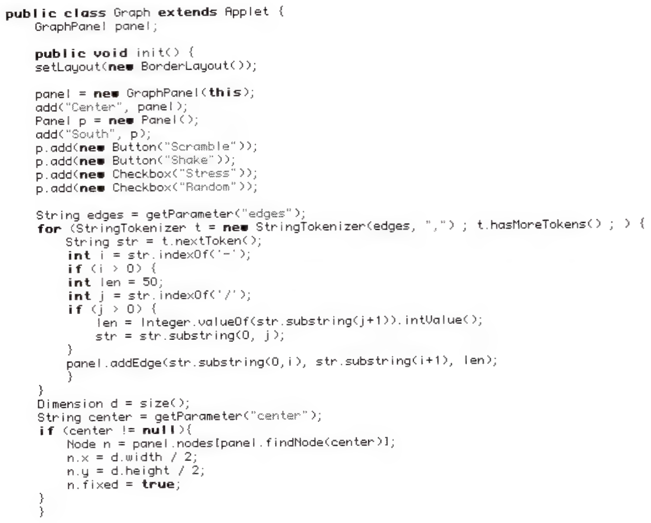
MW ★★★★
Macworld - May '96

orders: (800) 739-1616
email: orders@natural.com
web: www.roaster.com
international: (617) 556-2588

Natural Intelligence
www.natural.com, info@natural.com

Roaster is a trademark of Natural Intelligence, Inc. All rights reserved. Copyright © 1995-1996 Natural Intelligence, Inc. and its licensors. Macintosh is a registered trademark of Apple Computer, Inc. Mac is a trademark of Apple Computer, Inc. Java and Sun are trademarks of Sun Microsystems, Inc. All tradenames referenced are the trademark or registered trademark of their respective holders.

By this time, you're probably wondering what the Java source code looks like. I captured a portion of the code from the `Graph.java` source file, and it is shown in Figure 4.



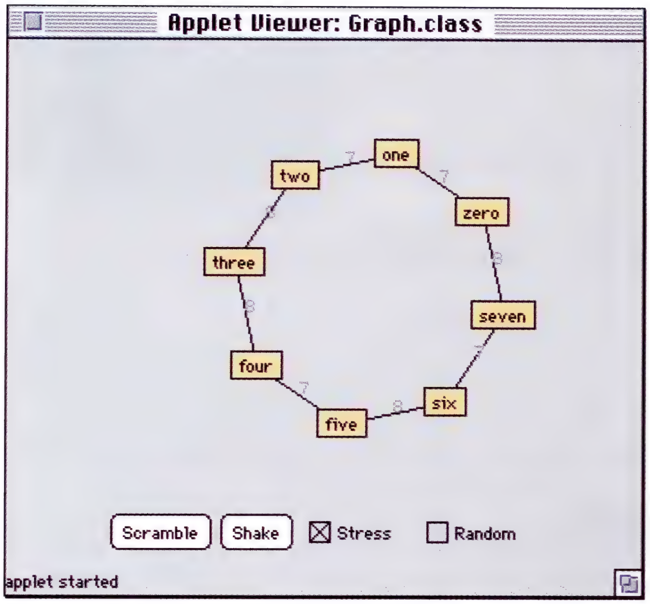
```
public class Graph extends Applet {
    GraphPanel panel;

    public void init() {
    setLayout(new BorderLayout());

    panel = new GraphPanel(this);
    add("Center", panel);
    Panel p = new Panel();
    add("South", p);
    p.add(new Button("Scramble"));
    p.add(new Button("Shake"));
    p.add(new Checkbox("Stress"));
    p.add(new Checkbox("Random"));

    String edges = getParameter("edges");
    for (StringTokenizer t = new StringTokenizer(edges, ",") ; t.hasMoreTokens() ; ) {
        String str = t.nextToken();
        int i = str.indexOf('-');
        if (i > 0) {
        int len = 50;
        int j = str.indexOf('/');
        if (j > 0) {
            len = Integer.valueOf(str.substring(j+1)).intValue();
            str = str.substring(0, j);
        }
        panel.addEdge(str.substring(0,i), str.substring(i+1), len);
        }
    }
    Dimension d = size();
    String center = getParameter("center");
    if (center != null){
        Node n = panel.nodes[panel.findNode(center)];
        n.x = d.width / 2;
        n.y = d.height / 2;
        n.fixed = true;
    }
    }
```

Figure 4. Portion of graph.java source code

The code shown in Figure 4 is just a sample of one function in the overall file. It shows how Symantec's built-in editor provides syntax coloring of keywords and character strings in the Java language. You can choose character styles and colors for language keywords, preprocessor directives, comments, string literals, and character constants. These choices are global and apply to these features for every language supported by the SPM.

One of the things that I really hate about Sun's Java code is the old-style statement formatting. I have campaigned for years against the old "Unix-style" statement formatting, and it seemed as though people were listening. More and more, code printed in magazines has been formatted for readability rather than for saving lines of code that took time to output on the old ASR 33 teletype printers. It really bothers me that Sun is creating a bad example that other Java programmers will tend to blindly follow. It isn't necessary to put the opening brace on the same line as the `if` statement, nor is it necessary to save lines by writing ugly code such as `} else {`. I just hope that programmers reading this review will think twice about the way that they format their own Java code. Please don't blindly follow Sun's lead.

That said, and having shown one example of the Graph Layout applet, I thought it would be useful to show the result of running the "example2.html" file, with the **Stress** checkbox selected. The source program does not need to be recompiled to use a different HTML file as input. Just open the "example2.html" file from within the Applet Viewer application (or from within your Internet browser, if you've installed the classes and HTML files where they belong). The result is shown in Figure 5.



applet started

Figure 5. Result of running example2.html

By running the example with the **Stress** checkbox checked, the nodes arrange themselves into a ring. The faint numbers on the lines connecting the nodes seem to be "weights" or "distances" that need to be preserved in this mode. Without the **Stress** checkbox checked, the nodes might arrange themselves into a number of different configurations. I've seen them in a figure-eight pattern.

MORE ABOUT JAVA

Although the Caffeine application doesn't support stand-alone applications, there is a lot that can be done with Java applets – as is amply demonstrated by the sophisticated examples. Sun's JDK includes the entire Java class library, including all of the AWT (Another Windowing Toolkit) classes, net (TCP/IP) classes, components (menubar, menu, container, scrollbar, window, dialog, panel, and so forth), as well as input/output classes, image classes, errors and exceptions, and many others. The Java class libraries are very well developed and seem to be mostly error-free. I've used the Windows version of Café to build a stand-alone application that has all of the elements of a standard application, except that it must still be executed with Sun's Java interpreter. This will undoubtedly be the case for the Mac version of Café when it is released.

The Java language is a kinder, gentler version of C++. There are no pointers in the new language, and applets are prevented from performing I/O on the user's machine. With the exception of the import of audio or graphic files, Java applets can be safely distributed on the net (at least that's the intention). Instead of requiring a "server-push" capability to provide dynamic Web pages, you'll be able to supply a Java applet that is transferred to the user's machine and executed by the Java-equipped Web browser (the released Mac version of the Netscape browser is still Java-challenged, but beta versions that offer Java support are available for download from Netscape's Web site). Java applets are also checked before they are executed to ensure that they don't try to wreak havoc on the user's machine.

The Windows version of the Netscape browser offers full Java applet execution. If you log on to Symantec's Café Web page, you'll see steam rising from the coffee cups that serve as bookends at the head of the page. This is accomplished by a Java applet that is downloaded to your machine automatically when you visit the page.

AND NOW BACK TO THE REVIEW

Symantec's SPM editor now includes drag-and-drop editing, proper opening and reading of files created with Unix, DOS, or Macintosh line-endings, and syntax coloring for all of the supported languages, including Java and PowerPC assembler files.

The Debugger has been improved to support deferred generation of debugging information. Applying this feature to files that you seldom debug can result in significant savings of disk space, as well as saving time during compilation.

The Visual Architect has been improved both by eliminating reported bugs and also by streamlining the user interface.

This release contains a set of highly interactive tutorials that guide you through the fundamental steps of using the Symantec development environment. Apple Guide files for four projects (Hello World, MiniEdit, Object Bullseye, and Beeper) are provided. These tutorials are accessible from the Guide menu whenever the Symantec project manager is running. In addition, there is a "Shortcuts & Tips" guide that is also accessible from the Guide menu. It provides useful information when using the Symantec Project Manager. The main page of the aforementioned guide is shown in Figure 6.

As you can see from the guide window in Figure 6, there is information on quite a number of topics. I chose the "Drag and drop" item and got the explanation shown in Figure 7.

The SPM now includes Apple's MrC and MrC++ translators for optimized compilation. You'll have to use the Custom Install option and choose the "Additional Drop-in Tools" category for these to be installed. In addition to the MrC and MrC++ translators, this category includes the Caffeine translator and the Sun SDK software, as well as Language Systems' LSPascal compiler. The new LSPascal compiler is still undergoing development; however, the version that is included in the current release is able to compile quite a number of applications, including the "NIH Image" image-editor demo program that is provided in the release. I understand that others have run into some problems with this version of the LSPascal compiler, so your mileage may differ.

In addition to the LSPascal translator, there is also a MacMindy translator that offers the ability to create Dylan applications for the Mac. Although the applications are interpreted, the MacMindy translator will give you the opportunity to play around with the Dylan language.

We Use
State-of-the-Art Tools
To Create Your Web Site

At JointSolutions Marketing, we have a full suite of advanced authoring and production tools to design innovative destinations on the World Wide Web.

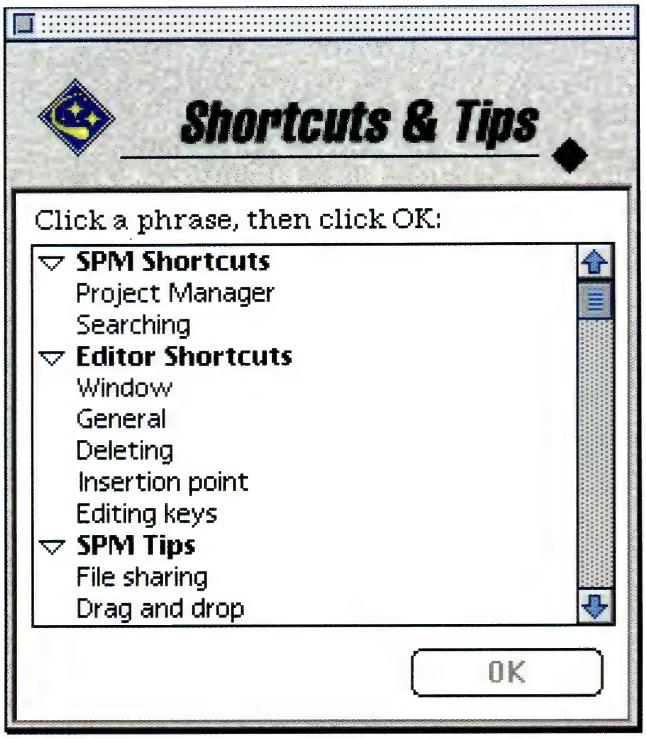
With experience working for clients like Apple Computer and SCO, we can handle your every need—from concept, through content, to completion. Large or small, we have the tools for your tasks.

We can even give you an address on our Web server, for hassle-free maintenance and updates.

So call today; our teams of technology specialists are standing by to assist you.

HTTP://WWW.JOINTSOLUTIONS.COM

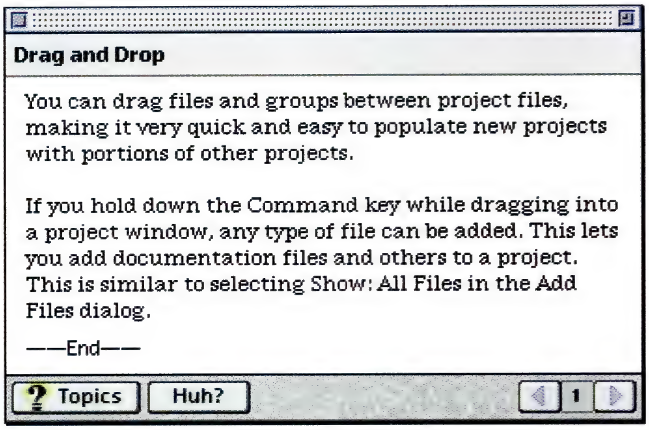
JOINT SOLUTIONS MARKETING
JointSolutions Marketing
Tel: (408) 471-1500
Fax: (408) 471-1511
http://www.jointsolutions.com



Click a phrase, then click OK:

- SPM Shortcuts
  - Project Manager
  - Searching
- Editor Shortcuts
  - Window
  - General
  - Deleting
  - Insertion point
  - Editing keys
- SPM Tips
  - File sharing
  - Drag and drop

Figure 6. Shortcuts & Tips guide



Drag and Drop

You can drag files and groups between project files, making it very quick and easy to populate new projects with portions of other projects.

If you hold down the Command key while dragging into a project window, any type of file can be added. This lets you add documentation files and others to a project. This is similar to selecting Show: All Files in the Add Files dialog.

——End——

Figure 7. Drag and Drop features

MORE ABOUT 68K AND POWER MAC DEVELOPMENT

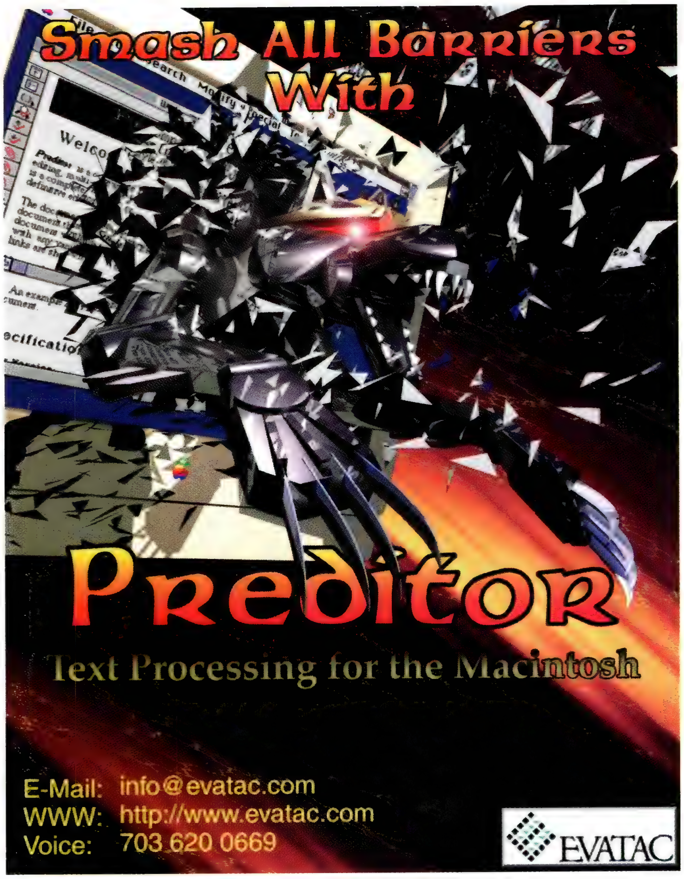
The SPM provides a unified environment for PowerPC and 68K application development. A single project can be configured to target both architectures, using named sets of project and translator options. This allows the use of the same source code *and* the same project file to build your application. You have a choice of building a PowerPC, standard 68K, or CFM (Code Fragment Manager) 68K application, library, or shared library. You can even

MAY THE SOURCE BE WITH YOU

Don't let the dark forces of ignorance defeat you. Right in this galaxy, you can tap into the source -- the free Consumer Information Catalog. It lists free and low-cost government publications on cosmic topics such as federal benefits, jobs, health, housing, educating your children, cars, and much, much more. So dispel the darkness and send for the source. Write today to Pueblo, Colorado for the free Consumer Information Catalog. Just send your name and address to :

Consumer Information Center
Department Source
Pueblo, Colorado 81009

A public service of this publication and the Consumer Information Center of the U. S. General Services Administration



E-Mail: info@evatac.com
WWW: http://www.evatac.com
Voice: 703 620 0669

choose to merge an already built 68K application into a PowerPC application, creating a fat application. The SPM itself is fat, as are the translators, the Visual Architect, and other tools.

The complete THINK Reference™ Viewer and databases are also included in this release. Although the references to the Macintosh toolbox aren't entirely up to date, THINK Reference is a must-have tool if you're programming with the Standard ANSI Libraries or the TCL. Apple's Macintosh Programmer's Toolbox Assistant doesn't cover those topics.

SYMANTEC GOODIES

Release 5 includes quite a number of additional "goodies" pertaining to the SPM. There is a "Compiler Toolkit" that explains how to host a new compiler within the SPM environment. The kit contains instructions and example projects. There's also information concerning external editors. If you have a personal editor preference, then you might be able to use it instead of the built-in SPM editor. Both Bare Bones Software's BBEdit and the Alpha shareware editor work well as external editors for the SPM. I've heard that Object Master also works in conjunction with the SPM, although there may be a problem with the SPM recognizing that files have been altered and need to be recompiled when that product is used.

The release also contains a folder full of scripts called "Do for Each Project," that perform operations such as "Remove Objects" on a set of projects. The Rainmaker Build Script provides an exhaustive log of the system features and memory usage during the build process. If you want to know everything about memory usage during compilation, then this script's for you.

An "SPM Color Manager" application provides the ability to change the SPM's default background and normal text colors. It's a FaceSpan application and comes with an installer and the FaceSpan extension.

There's source code too. The `CMoviePane` class offers the ability to show QuickTime movies in a pane. The `CColorBackground` and `CColorBorder` classes offer the ability to use color in the background or border of a window or dialog pane. And the `CDoubleText` class operates like the `CIntegerText` class, except that it handles text strings that represent "real" numbers. All of the foregoing classes are experimental and are not supported by Symantec, but are provided as extra "goodies" for you to use.

There are many more goodies than I have space to mention. You'll just have to scour the CD to see the wealth of programs and data that the release has to offer.

THIRD-PARTY GOODIES

The release also contains a wealth of third-party shareware and demo applications. There are offerings from Bare Bones Software (a demo of the BBEdit v3.5 editor), NeoLogic (full source and demos for the neoAccess cross-platform object-oriented database engine), demo versions of a variety of utilities, including Resorcerer, Script Debugger, Developer VISE, VOODOO, CMaster v2.0.3, Tools Plus v2.1.6, and many others. There are also free versions of ProFont (a font created especially for displaying and printing code) and ShrinkWrap (an application that lets you mount disk images on your desktop, instead of having to install system versions or other software from floppy disks).

THE BOTTOM LINE

The Symantec C++ for Power Macintosh product retails for $399. Those of us who received this product after having purchased Symantec C++ v8.0 will see that renewal of the subscription for the next three issues will cost $149.95. CodeWarrior users can also trade up to this product for only $149.95, which is $50 less than renewing their CodeWarrior subscription.

There's very little not to like about the product. I like the smaller footprint and the increased efficiency of this release. I also like the ability to develop Java applets within the integrated environment. The built-in editor, compilation, and viewing features are quite efficient. I think you'll like this product too.

WEBTECH™

By Mathieu Lafourcade and Gilles Sérasset

CGIs in Lisp

A Web dictionary server

INTRODUCTION

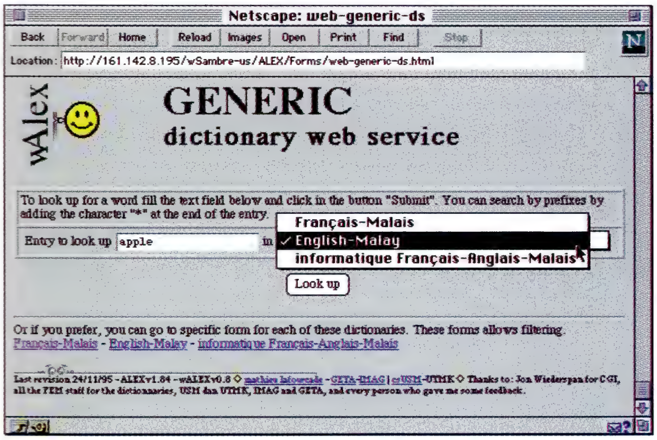
A paper dictionary is fine for reading comfort, versatility, etc., but updating of the data is not possible (apart from buying a new dictionary!), and the amount of data may be smaller than an electronic version. A resident electronic dictionary allows cut and paste, and updates are generally possible; but, like the paper dictionary, you must carry it along (in your hard disk). The Web-based dictionary doesn't take any space on your hard disk; the memory footprint is that of your Web browser. On the other hand you do rely on the availability of the network and/or server.

This article describes how a Web server offering multilingual dictionary services was set up by integrating three components: MacHTTP, AppleScript, and Macintosh Common Lisp (MCL). The dictionary server is constructed from a dictionary application, ALEX, written with MCL. This application can be remotely (and facelessly if needed) controlled with AppleScript. Given this design, the connection with MacHTTP is painless. The result is a very cost-effective way of providing a large linguistic resource on the Web.

Our developmental approach was to come up with something functional as fast as possible and to enhance the solution afterward. We had to compromise between functionality delivered to the user, performance, and the technical implications for the developer.

We first describe a typical user session; you can try this out yourself by pointing to the URL from your own browser (you can see the URL in Figure 1). The general architecture of the server is then presented, followed by some technical details about the three components.

Note: The included sources have often been trimmed to keep this article as short as possible – for instance, the error handling is generally not included.



Netscape: web-generic-ds

Location: http://161.142.8.195/wSambre-us/ALEX/Forms/web-generic-ds.html

GENERIC
dictionary web service

To look up for a word fill the text field below and click in the button "Submit". You can search by prefixes by adding the character "*" at the end of the entry.

Entry to look up apple in

Français-Malais
✓ English-Malay
informatique Français-Anglais-Malais

Or if you prefer, you can go to specific form for each of these dictionaries. These forms allows filtering.
Français-Malais - English-Malay - informatique Français-Anglais-Malais

Last revision 24/11/95 - ALEXv1.84 - wALEXv0.8 ◇ mathieu lafourcade - GETA-IMAG | cs USM-UTMK ◇ Thanks to: Jon Wiederspan for CGI, all the FEM staff for the dictionnaries, USM dan UTMK, IMAG and GETA, and every person who gave me some feedback.

Figure 1. Initial Web page of the server

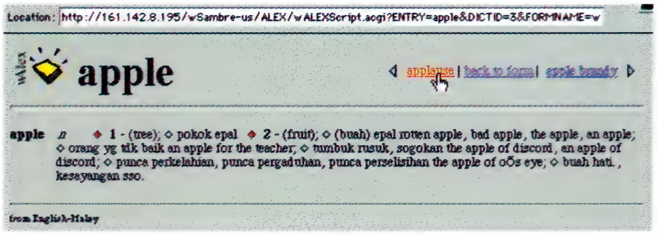
Mathieu Lafourcade works for the Study Group for Machine Translation (GETA) in Grenoble, France, but is currently enjoying several months in Malaysia under cover of a French-English-Malay dictionary development project at UTMK-USM (University of Sciences, Malaysia) at Penang. Besides expecting a baby in the coming months, he is working on software engineering for lingware. You can reach him at Mathieu.Lafourcade@imag.fr or lafourca@cs.usm.my.

Gilles Sérasset, who also works for GETA, is stuck in the cold weather of the Alps, and looking for any opportunity to go sunbathing on the Batu Ferringhi beaches of Penang. You can reach him at Gilles.Serasset@imag.fr.

A Typical User Session

The Web page displays a form (Figure 1). The user can choose a dictionary, enter a word to look up, and click on the "Look up" button. It is possible to terminate the candidate word with an asterisk; in this case the first entry which starts with the text entered is returned.

A new page appears displaying the information associated with the word entered (Figure 2). If the entry is not found, a (friendly) error message is given. Beside the title (corresponding to the word looked up), the user has access to links to the previous word (if any), to the form (just like pressing the "Back" button of the Web client), or to the next word (if any). This way, the user can walk through the entire dictionary.

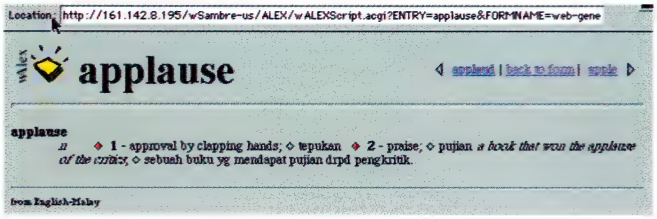

Location: http://161.142.8.195/wSambre-us/ALEX/wALEXScript.acgi?ENTRY=apple&DICTID=3&FORMNAME=w

apple *n* ◆ **1** - (tree); ◇ pokok epal ◆ **2** - (fruit); ◇ (buah) epal rotten apple, bad apple, the apple, an apple; ◇ orang yg tdk baik an apple for the teacher; ◇ tumbuk rusuk, sogokan the apple of discord, an apple of discord; ◇ punca perkelahian, punca pergaduhan, punca perselisihan the apple of oÕs eye; ◇ buah hati, kesayangan sso.

from English-Malay

Figure 2. The data displayed for the entry "apple"; the source of this page is given in Listing 10

For example, the user can click on the previous entry "applause" and obtain the corresponding page (Figure 3). The user can continue to scan the dictionary or can return to the form and choose an other entry or dictionary.


Location: http://161.142.8.195/wSambre-us/ALEX/wALEXScript.acgi?ENTRY=applause&FORMNAME=web-gene

applause
n ◆ **1** - approval by clapping hands; ◇ tepukan ◆ **2** - praise; ◇ pujian *a book that won the applause of the critics*, ◇ sebuah buku yg mendapat pujian drpd pengkritik.

from English-Malay

Figure 3. The web page for the entry "applause"

The HTML Form

The form used as an interface between user and server, the starting point of a session, is a plain HTML page. The link between this page and the application handling the action is made in the form declaration. The standard `POST` method is used for sending the message from the server to the application, wAlex.acgi.

We assume you are familiar with HTML (if not, check out the numerous informative pages available on the Web). The `<FORM>` tag specifies a form, which may include various items, such as text input (text field) and select input (menu). Notice that many less common characters (in English at least) are represented a specific way (for example the "ç" as `"&ccedil"`), and this is the beginning of big problems when you decide to deal with other languages, like French for instance.

Listing 1:
HTML Source of Figure 1

The dictionary lookup form

All the cosmetic and informative stuff at the beginning of the page.

```
<TITLE>web-generic-ds</TITLE>
<TABLE CELLPADDING=3 WIDTH="100%">
<TR>
  <TD ALIGN=LEFT>
  <img ALIGN=middle src="http://161.142.8.195/wSambre-
us/ALEX/Gifs/wAlex-vert.GIF" WIDTH=25>
  <img ALIGN=middle src="http://161.142.8.195/wSambre-
us/ALEX/Gifs/wAlexVLogo.GIF">
  </TD>
  <TD ALIGN=LEFT>
<H1><FONT SIZE=+7>GENERIC</FONT><BR> dictionary web
service</H1>
 </TD>
</TR>
</TABLE>
```

The form definition starts here.

```
<FORM ACTION="http://161.142.8.195/wSambre-
us/ALEX/wAlex.acgi" METHOD=Post>

<TABLE border UNITS=PIXELS CELLPADDING=2 WIDTH="100%">
<TR>
  <TD>
  <P>To look up for a word fill the text field below and
click in the button "Submit".
You can search by prefixes by adding the character "*" at the
end of the entry.<BR>
  </TD></TR>
<TR>
  <TD ALIGN=CENTER>Entry to look up <INPUT TYPE="text"
NAME="Entry" VALUE="" MAXLENGTH=150>
in
<SELECT NAME="DictID">
<OPTION VALUE="2">Fran&ccedil;ais-Malais
<OPTION VALUE="3">English-Malay
<OPTION VALUE="1">informatique Fran&ccedil;ais-Anglais-Malais
</SELECT>
  </TD></TR>
</TABLE>
```

The Submit button is defined here.

```
<P><CENTER><INPUT TYPE="Submit" VALUE="Look up"></CENTER>
```

One hidden item giving the name of the form we are in.

```
<INPUT TYPE=HIDDEN NAME="FormName" VALUE="web-generic-
ds.html">
```

The form definition ends here.

```
</FORM>
```

Some links to specific dictionary forms are presented here.

```
<HR>
Or if you prefer, you can go to specific form for each of
these dictionaries.
These forms allows filtering. <BR>
<A HREF="http://161.142.8.195/wSambre-us/ALEX/Forms/web-FEM-
ds.html">Fran&ccedil;ais-Malais</A> -
<A HREF="http://161.142.8.195/wSambre-us/ALEX/Forms/web-EM-
ds.html">English-Malay</A> -
<A HREF="http://161.142.8.195/wSambre-us/ALEX/Forms/web-
FEMterm-ds.html">informatique Fran&ccedil;ais-Anglais-
Malais</A>
```

The signature ends the page.

```
<P><IMG ALIGN = MIDDLE src="http://161.142.8.195/wSambre-
us/GIFgeneral/line.GIF"></br>
<FONT SIZE = 1> Last revision 24/11/95 - ALEXv1.84 -
wALEXv0.8  <IMG SRC="http://161.142.8.195/wSambre-
us/ALEX/Gifs/yellow-diamond.GIF">
<A HREF="http://161.142.8.195/wSambre-us/STAFF/ML-
```

```
us.html">mathieu lafourcade</A> -
<A HREF="http://imag.fr/IMAG/GETA.html">GETA</A>-<A
HREF="http://imag.fr/">IMAG</A> |
<A HREF="http://www.cs.usm.my/">csUSM</A>-<A>UTMK</A> <IMG
SRC="http://161.142.8.195/wSambre-us/ALEX/Gifs/yellow-
diamond.GIF">
 Thanks to: Jon Wiederspan for CGI, all the FEM staff
for the dictionaries, USM dan UTMK, IMAG and GETA, and every
person who gave me some feedback.
</FONT><P>
```

Opening Pandora's Box

How does the information get from the browser to the server to the Dictionary Service? In form-based queries, there are two methods (`POST` and `GET`) for submitting the data to the gateway script. We use only `POST`, as recommended by NCSA. `POST` (unlike `GET`) is not limited in the amount of information you can pass with a form.

A gateway script is needed on the server side to decode these arguments, process the query, and return an answer. In our case we have MacHTTP (the server application) and our MCL-based application, ALEX (which serves dictionary data). We have tested two different architectures for connecting MacHTTP to ALEX.

In the first solution, we transformed ALEX, our MCL appplication, itself into an acgi script. The second solution uses a small AppleScript applet to make the glue between MacHTTP and ALEX. Both methods have their own advantages and drawbacks.

ALEX (MCL) As a CGI Application

In the first solution MacHTTP interacts directly with ALEX, our MCL application (Figure 4). To make MacHTTP believe it is an acgi script, the name of the application must end with "`.acgi`". When MacHTTP receives a query from a WWW client requesting a file whose suffix mapping is "`.acgi`" (an executable application), MacHTTP will attempt to communicate with it via a custom Apple event of class `'WWWΩ'` and ID `'sdoc'`. We want the received request to be handled by ALEX; hence, our application has to listen (at least) to this particular Apple event. For this solution, we also have to change the name of the form action in the HTML file to `wAlex.acgi` (which is the name of our MCL application).

MCL directly supports the four required Apple events sent by the Finder: Open Application, Open Documents, Print Documents, and Quit Application. In addition, MCL provides a very easy way to support other Apple events by writing and installing new Apple event handlers.

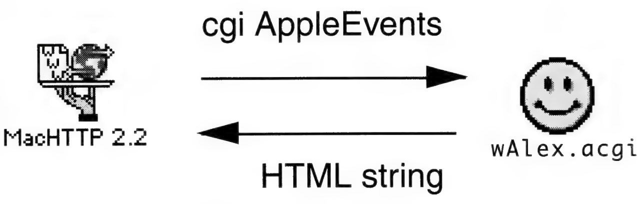


Figure 4. MacHTTP communicates directly with ALEX transformed into an ACGI application

Handling Apple events

An Apple event handler is a generic function specialized on the `application` class. Hence it is possible to customize the behaviour of the predefined Apple event handlers by specializing the `application` class. This is not necessary for our purpose. However, we have to write a new handler for the CGI event.

As MCL bypasses the Apple Event Manager's dispatch routine, you don't have to bother with how your function is called by the Apple Event Manager. Lisp takes care of dispatching and run-time error checking. If no error occurs, your handler should simply return in a normal way. The returned value is ignored. If an error occurs, your handler should signal an `appleevent-error` condition, with an error number and an optional error string.

An Apple event handler has four arguments: the *application* (always the value of the global variable `*application*`), the *Apple event* (you generally don't have to look at the record structure directly), the *reply* (another Apple event record, provided by the Apple Event Manager) and an optional handler *reference constant* (any Lisp object that can be used e.g. to distinguish two different installations of the same handler; this will be ignored for our purpose). Look at Listing 4.

Decoding arguments

MacHTTP provides a lot of information (client address, user name and password, arguments of the request, etc.). MacHTTP's documentation will give you the complete set of parameters and their four-byte codes.

To get the value associated with each parameter, you have to use the `ae-get-parameter-char` function. You specify the AppleEvent object and the parameter, and the function returns a string containing the required value.

In our case, we assume that the client issues a `POST` method (the `GET` method case can be easily deduced from this code). Hence, the arguments we need are the post arguments, accessible via the `'post'` four-byte code.

The `ae-get-parameter-char` function returns a string consisting of all the variables of the form, and their values appended together. This format is not directly usable; hence, we will transform it into a list of variable-value pairs. For instance, the argument string:

```
"ENTRY=apple&FORMNAME=web-generic-ds&DICTID=3"
```

will be transformed into:

```
(:DICTID "3"
 :FORMNAME "web-generic-ds"
 :ENTRY "apple")
```

To transform this kind of string, we use one of our publicly available packages called `geta-strings` (ftp://ftp.digitool.com//pub/MCL/contrib/geta-strings-1.3.sea.hqx).

Listing 2:
wAlex.acgi 1/3 (MCL Application)

Decoding POST arguments

```
(defun create-keywords-from-args (args)
;; segment the string with '&' as a separator
 (let ((mylist (geta-strings::list-words args "&"))
       pair
       result)
;; for each part, segment with '=' as a separator and construct the result
  (dolist (lv-pair mylist)
    (setf pair (geta-strings::list-words lv-pair "="))
    (push (second pair) result)
    (push
     (read-from-string
      (concatenate 'string ":" (first pair))) result)
      )
    result))
```

This list will be simply used as a list of arguments for a function accepting all the variables defined in the HTML form as keys. Hence, we rely on the standard Lisp binding mechanism to bind each value to its variable. For our purpose, we ask the appropriate dictionary for an HTML version of the entry structure.

Listing 3:
wAlex.acgi 2/3 (MCL Application)

Getting the corresponding entry

```
(defun get-walex-entry  (&key dictid
                          dicttype
                          formname
                          entry
                          &allow-other-keys)
;; Now, we can directly use each variable to query our dictionary
  (let ((dict (get-dictionary dictid dicttype)))
    (findAndTransmuteItem
      dict
      entry :output-format :html-1))
  )
```

Replying

MacHTTP expects the Apple event reply's direct parameter to contain an HTTP/1.0 header and HTML text that will be transmitted to the client. Hence, we have to construct an appropriate answer. The HTML/1.0 header will be built as a global variable:

```
;; Define the http 1.0 header (beware of crlf)
(defvar *crlf*
  (format nil "~A~A" #\Linefeed #\Newline))
(defvar *http-10-header*
  (format nil "HTTP/1.0 200 OK ~A~
               Server: MacHTTP ~A~
               MIME-Version: 1.0 ~A~
               Content-type: text/html ~A~A"
          *crlf* *crlf* *crlf* *crlf* *crlf*))
```

The reply itself will be returned by the `findAndTransmuteItem` function. Hence, we just have to put the header and answer in the Apple event reply's direct parameter with the `ae-put-parameter-char` function.

Listing 4:
wAlex.acgi 3/3 (MCL Application)

Defining and installing the Apple event handler

```
(defmethod my-ae-handler ((a application)
                          theAppleEvent
                          reply
                          handlerRefcon)
;; We don't use the handlerRefcon argument
  (declare (ignore handlerRefcon))
;; Just get the post parameter, decode it, handle it and reply
  (let* ((post-args
           (create-keywords-from-args
             (ae-get-parameter-char theAppleEvent
                                    :|post| t)))
         (the-answer
          (apply #'get-walex-entry post-args)))
;; Build the reply and put it into the Apple event reply's direct parameter
    (ae-put-parameter-char reply #$keyDirectObject
                           (format nil "~A ~A"
                                   *http-10-header*
                                   the-answer))
))
(install-appleevent-handler :|WWWΩ| :|sdoc| #'my-ae-handler)
```

AN APPLESCRIPT APPLET AS GLUE

In this second solution (Figure 5) we use an AppleScript applet to make the glue between MacHTTP and ALEX (MCL). As in the previous solution, the applet is named wAlex.acgi. It is now up to the applet to decode the argument passed by MacHTTP. This time the decoding is done thanks to several Scripting Extensions.

The Tokenize function allows you to transform a string with separating characters into a list. For example, from `"A;list;of;word"` you will get `{"A" "list" "of" "words"}` if you take ; as the separator. DePlus does the same, but with the + character (more specific and more efficient). Encoding a URL consists of replacing any problematic characters with their `%xx` counterpart (for example, the space is replaced by `%20`). Decoding a URL is the reverse process. DecodeURL_8bits does more by also handling the nightmarish problem of accents (and diacritics in general).

After being activated by MacHTTP, the wAlex.acgi applet calls specific AppleScript Programming Interface (ASPI) procedures of the dictionary server application ALEX (MCL).

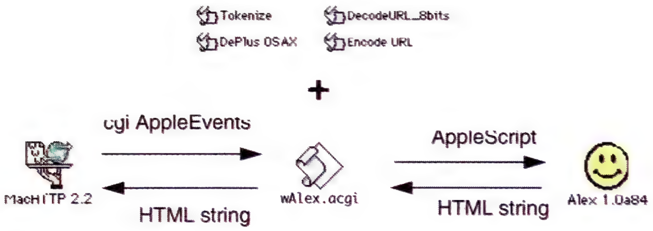


Figure 5. MacHTTP communicates indirectly with ALEX through an AppleScript applet

The communication between MacHTTP and wAlex.acgi is asynchronous. This means that MacHTTP is not waiting for the script to reply. But the communication is synchronous between wAlexScript and ALEX. As we didn't deal with time-out in our script, the default value of 60 seconds applies. ALEX has to reply within one minute (no user will stand a much longer delay anyway).

This solution is well adapted to our needs, as we would like to add more dictionaries that are not necessarily managed by our MCL application. With this solution, we need very little change to the AppleScript applet to integrate new tools.

The script

We use the `POST` method in our forms or in the links associated with the previous and next entries. In the entry point, the dispatching to the appropriate function is handled on the basis of the `method` argument.

Notice that for an AppleScript applet, when properties are set outside of the event handlers, they are "remembered" across launches. We want the script to be continuously open.

Listing 5: wAlex.acgi (AppleScript Applet)

Properties and CGI entry point

```
-- This code comes from Jon Wiederspan and has been modified to fit our purposes.
-- Some properties are defined for making it easier to read.
property crlf : (ASCII character 13) & (ASCII character 10)
property http_10_header : "HTTP/1.0 200 OK" & crlf & "Server:
MacHTTP" & crlf & "MIME-Version: 1.0" & crlf & "Content-type:
text/html" & crlf & crlf

property theApp : application "Alex"

-- The main entry of our script.
on «event WWWΩsdoc» path_args ¬
  given «class kfor»:http_search_args, «class
post»:post_args, «class meth»:method, «class
addr»:client_address, «class user»:username, «class
pass»:|password|, «class frmu»:from_user, «class
svnm»:server_name, «class svpt»:server_port, «class
scnm»:script_name, «class ctyp»:content_type

-- Decode the Post arguments.
  try
    – Tokenizing our arguments means creating a list (or arguments) form a string
    set tokenized_pa to tokenize post_args with delimiters
{"&"}

-- Save the original AppleScript text item delimiters.
    set oldDelim to AppleScript's text item delimiters

-- Use "=" to delimit the pairs.
    set AppleScript's text item delimiters to {"="}

-- Retrieve our arguments (it is positional and thus not very nice).
    set theEntry to (Decode URL
      (dePlus (last text item of item 1 of tokenized_pa)))
    set theBaseID to (Decode URL
      (dePlus (last text item of item 3 of tokenized_pa)))

-- Restore the original AppleScript text item delimiters.
    set AppleScript's text item delimiters to oldDelim

-- Returns the result (we don't use theFrom for simplicity)
    return http_10_header &
      getAlexEntryWithID(theBaseID, theEntry, post_args)

-- Error handling (not detailed).
  on error msg number num
    ...
  end try
end «event WWWΩsdoc»
```

For our purpose we consider only two arguments (although some others could be useful later on): the word to look up, and a reference to the dictionary. The entry is strictly the string given by the user. The base ID is strictly the selected option of the selected item (in the list of available dictionaries). You can see that in fact we pass the whole argument string along with the two decoded argument. We use this technique for handling optional arguments. In this case the string should be decoded by ALEX (exactly the same way it is done in our first solution). The value of `post_args` giving rise to Figure 2 is `"ENTRY=apple&DictID=3"`. You can see from Figure 1 what settings in the form this refers to.

AND THE WINNER IS...

The main advantage of the first solution is that the direct interaction between MacHTTP and MCL doesn't make use of AppleScript. The integration is lighter, and thus there is better overall performance in speed and robustness. But you have to delve into the decoding of AppleEvents.

The second solution is technically simpler, and you can very easily change the Lisp interface without changing the overall processing. You can also use other applications to manage dictionaries accessible by the same form. The choice of the target application is done by the applet.

For our actual implementation we retained the second solution, but the choice depends on the goals you assign to your application.

One last question can arise: where we should decode the arguments. We have presented two radically different ways of decoding, with either MCL or the AppleScript applet doing the job. Actually, in the current version of our server, we used a mixture of the two approaches. The AppleScript is used, but decodes only certain arguments (the dictionary name), and always passes the whole argument string to MCL. Then, MCL itself decodes the string with the help of the code presented in listings 2 and 3.

ALEX

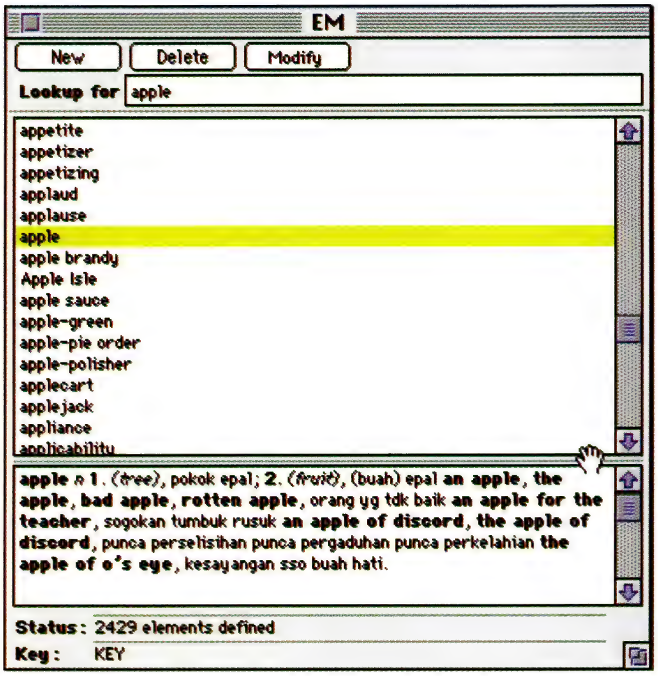
ALEX, the application that serves dictionary information, has been developed with MCL. This tool can be used as an end-user tool with a user interface, or can be remotely and facelessly controlled with AppleScript.

Overview: a small session

ALEX is a small tool allowing a user to open and browse dictionaries. The interface, although minimal, seems quite efficient for the assigned purpose. This tool has been developed as a part of research on the kind of interface needed (or desirable) for structured dictionaries.

The user can scroll through the entry list (not implemented with the list manager, which is limited to around 4000 items), or can make simple active lookups. By active lookup we mean that as the user enters characters into the text box at the top of the window the first entry starting with those characters is selected.

What makes ALEX somewhat different from other tools is that all the entries are kept structured and displayed on the fly, allowing the user to specify some filtering process. It is possible to ask for several kinds of formatting. Another aspect, still under investigation, is the complete scriptability of ALEX.



Lookup for apple

appetite
appetizer
appetizing
applaud
applause
apple
apple brandy
Apple Isle
apple sauce
apple-green
apple-pie order
apple-polisher
applecart
applejack
appliance
applicability

apple *n* 1. *(tree)*, pokok epal; **2.** *(fruit)*, (buah) epal **an apple, the apple, bad apple, rotten apple**, orang yg tdk baik **an apple for the teacher**, sogokan tumbuk rusuk **an apple of discord, the apple of discord**, punca perselisihan punca pergaduhan punca perkelahian **the apple of o's eye**, kesayangan sso buah hati.

Status: 2429 elements defined
Key: KEY

Figure 6. An ALEX window displaying the entry "apple" of the English-Malay dictionary

Architecture

The implementation of ALEX is based on several layers, all implemented with MCL. MCL relies on a Dictionary Object Protocol. This manager allows the developer to define multi-key dictionaries and manipulate them. The DOP itself is based on Wood, a persistent Object-Oriented Database developed by Bill St. Clair (ftp://ftp.digitool.com/pub/mcl/contrib/wood/). The persistency allows a minimal memory footprint – a dictionary is never completly loaded into memory. ALEX can be considered as a reduced user interface for the DOP, allowing remote control by AppleScript.

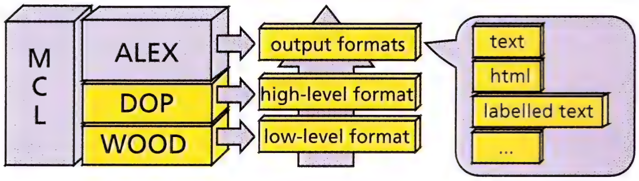


Figure 7. Layers of the actual implementation of ALEX. The Wood layer stores the low-level format of the dictionary entry. The DOP dynamically decodes each entry in a high-level format. From this high-level format, ALEX produces any output format.

AppleScript Programming Interface (ASPI)

The `do script` AppleScript command is a universal command for controlling ALEX. This is very practical for debugging and trying new ideas; you just have to throw Lisp code at ALEX through AppleScript. The dark side is that anyone can do anything. In order for this to work, don't forget to load the `eval-server.lisp` file located in the MCL Examples folder with a `(require :eval-server)` form.

We defined some AppleScript commands that can do specific actions: opening and closing a dictionary, looking up, inserting and deleting entries, etc. The two transmute functions are the ones used in the HTML code.

Listing 6:
ASPI

getAlexEntryWithID

```
-- Get the data in the HTML format with the id of the base.
on getAlexEntryWithID(id, entry, args)
  tell TheApp
    do script ("(get-alex-item (alex::base
      (alex::get-dict-handler \"" & id &
      "\"  :id)) " & "\"" & entry & "\"" &
      " :output-format :html-1 & :args & "\"" args & "\")")
  end tell
end getAlexEntryWithID
```

As you can see, scripting our MCL application consists simply of building strings that, once given to the Lisp interpreter, will produce some effect. Yes, you can do Lisp from AppleScript! This is simple and neat. Of course, you cannot ask an end-user to write some Lisp code from AppleScript; this is why you have to provide what we call an AppleScript Programming Interface (ASPI). It is a set of AppleScript commands that will control your application. The `getAlexEntryWithID` function is one of the functions of ALEX's ASPI.

Many of the functions come in two flavors, with dictionary access by ID or by name. For example, `getAlexEntryWithID` has its `getAlexEntryWithName` counterpart. Some of them are documented in Listing 7.

Listing 7:
Other ASPI Useful for ALEX

openAlexDictionary

```
-- Open a base from a given filename. Return an index dictionary or 0 in case of error.
on openAlexDictionary(filename)
  tell theApp
    return do script ¬
      ("(alex::open-object :alex-base :return-index t :file
        (pathname \"" & filename & "\"))")
  end tell
end openAlexDictionary
```

getAlexDictionaryName

```
-- Return the dictionary name based on the dictionary id.
on getAlexDictionaryName(id)
  tell theApp
    return do script ("(alex:name (alex::base
      (alex::get-dict-handler \"" & id & "\"  :id)))")
  end tell
end getAlexDictionaryName
```

deleteAlexEntryWithID

-- Deletes an entry (if defined) based on the dictionary id.

```
on deleteAlexEntryWithID(id, entry)
  tell theApp
    do script ("(delete-item (alex::base
      (alex::get-dict-handler \"" & id & "\"  :id)) :key "
      & "\"" & entry & "\"" & ")")
  end tell
end deleteAlexEntryWithID
```

Take some time to document your ASPI, and you have achieved the same result as developing your AppleEvents suite, but with less effort and time. One main difference from the more classical method is that your application has no AppleScript dictionary and you cannot use clauses.

Dictionary data structure

The entries are kept in a specific format in the dictionary. This format has been designed to compromise between file space and decoding time. For our dictionary a DOP base is linked to a text file which stands for a text buffer. The low-level format is basically a vector of index-length pairs pointing to this buffer. When an item is found, it is dynamically decoded into a high-level format (example in Listing 10).

The `get-alex-item` function of ALEX (called by `getEntryWithID`) basically retrieves one dictionary item and calls `transmute-entry`, which transforms the high-level format into any output format specifed by the developer. We take advantage of the generic function mechanism of CLOS (Common Lisp Object System) in MCL to have an easily extensible collection of specific formats. The selection of the appropriate method is done on the basis of specialised arguments of the function, basically the type of the current item and the desired output format (cf. Listing 9).

Listing 8: Two CLOS Headers for Plain Text and HTML Formatting

transmute-entry

First method for of text output (code not detailed).

```
(defmethod transmute-entry (entry
                   (entry-type (eql :en-entry))
                   (output-format (eql :text)) )
  ...)
```

Second method for one kind of HTML output (code not detailed).

```
(defmethod transmute-entry (entry
                   (entry-type (eql :en-entry))
                   (output-format (eql :html-1)) )
  ...)
```

Listing 9: High-level Format of ALEX Items

High-level format for "apple"

The high level format of ALEX is a (kind of) property list. The first item is always the type of the entry. The rest of the structure is made of an open list composed of a type and one or several string values.

```
(:EM-ENTRY (:ENTRY "apple") (:C "n") (:N "1") (:G "(tree)")
(:ME "pokok epal") (:N "2") (:G "(fruit)") (:ME "(buah)
epal") (:P "an apple") (:P "the apple") (:P "bad apple") (:P
"rotten apple") (:MPE "orang yg tdk baik") (:P "an apple for
the teacher") (:MPE "sogokan") (:MPE "tumbuk rusuk") (:P "an
apple of discord") (:P "the apple of discord") (:MPE "punca
perselisihan") (:MPE "punca pergaduhan") (:MPE "punca
perkelahian") (:P "the apple of o's eye") (:MPE "kesayangan
sso") (:MPE "buah hati."))
```

Requested output format

The code produced by ALEX for the entry "apple" is given in listing 10 (the code has been slightly edited for readability). Besides the cosmetic features of the page, the most interesting part of the listing is the handling of the "previous" and "next" entries.

Listing 10: Source of Figure 2

HTML "apple"

Beginning of the header of the page and control part for access to the previous and next entries.

```
<TABLE CELLPADDING=0 WIDTH="100%"><TD><IMG
SRC="http://161.142.8.195/wSambre-us/ALEX/Gifs/walex-
vert.GIF" width=10> <IMG ALIGN=Bottom
SRC="http://161.142.8.195/wSambre-
us/ALEX/Gifs/stack.GIF"><b><FONT SIZE = +6> apple </FONT></B>
<TD ALIGN=Right><TABLE border=0>
```

The left arrow and the previous word. Here (and only here) the GET method is used in the urls (notice the ? character).

```
<TD VALIGN=CENTER>
<A HREF="http://161.142.8.195/wSambre-
us/ALEX/wALEXScript.acgi?ENTRY=applause&FORMNAME=web-generic-
ds.html&DICTID=3"><IMG SRC="http://161.142.8.195/wSambre-
us/ALEX/Gifs/left-arrow.GIF" BORDER=0></A></TD>
<TD VALIGN=CENTER><A HREF="http://161.142.8.195/wSambre-
us/ALEX/wALEXScript.acgi?ENTRY=applause&FORMNAME=web-generic-
ds.html&DICTID=3"> applause</A>
</TD>
```

The link back to the form.

```
<TD VALIGN=CENTER> | <A HREF="http://161.142.8.195/wSambre-
us/ALEX/Forms/web-generic-ds.html">back to form</A> | </TD>
```

The right arrow and the next word.

```
<TD VALIGN=CENTER><A HREF="http://161.142.8.195/wSambre-
us/ALEX/wALEXScript.acgi?ENTRY=apple+brandy&FORMNAME=web-
generic-ds.html&DICTID=3">apple brandy</A></TD>
<TD VALIGN=CENTER><A HREF="http://161.142.8.195/wSambre-
us/ALEX/wALEXScript.acgi?ENTRY=apple+brandy&FORMNAME=web-
generic-ds.html&DICTID=3"><IMG
SRC="http://161.142.8.195/wSambre-us/ALEX/Gifs/right-
arrow.GIF" BORDER=0></A></TD></TABLE></TABLE>
```

The informative part of this page.

```
<HR SIZE=4><DL COMPACT>
<DT><B>apple</B>
<DD><I>n</I><IMG HSPACE=8 SRC="http://161.142.8.195/wSambre-
us/ALEX/Gifs/space.GIF"> <IMG HSPACE=6
SRC="http://161.142.8.195/wSambre-us/ALEX/Gifs/red-
diamond.GIF">
<B>1</B> -  (tree); <IMG SRC="http://161.142.8.195/wSambre-
us/ALEX/Gifs/cyan-diamond.GIF"> pokok epal <IMG HSPACE=6
SRC="http://161.142.8.195/wSambre-us/ALEX/Gifs/red-
diamond.GIF">
<B>2</B> - (fruit); <IMG SRC="http://161.142.8.195/wSambre-
us/ALEX/Gifs/cyan-diamond.GIF"> (buah) epal  rotten apple,
bad apple, the apple, an apple; <IMG
SRC="http://161.142.8.195/wSambre-us/ALEX/Gifs/cyan-
diamond.GIF"> orang yg tdk baik  an apple for the teacher;
<IMG SRC="http://161.142.8.195/wSambre-us/ALEX/Gifs/cyan-
diamond.GIF"> tumbuk rusuk, sogokan  the apple of discord, an
apple of discord; <IMG SRC="http://161.142.8.195/wSambre-
us/ALEX/Gifs/cyan-diamond.GIF"> punca perkelahian, punca
pergaduhan, punca perselisihan  the apple of o's eye; <IMG
SRC="http://161.142.8.195/wSambre-us/ALEX/Gifs/cyan-
diamond.GIF"> buah hati., kesayangan sso.</DL>
```

Information to remind the user in which dictionary he is looking.

```
<HR><FONT SIZE=1>from English-Malay</FONT>
```

Note that the space character in "apple Brandy" has been replaced by a `'+'`. This has been done at the HTML page

TestTrack

Bug Tracking the Macintosh Way

1-2-3 Start Tracking
TestTrack is ready to use right out of the box—simply install, add a few users, and start tracking. It's that easy.

TestTrack is more than an easy-to-use bug tracking program—it's a powerful quality control tool for busy software development teams:

Automate TestTrack automates the tedious and error-prone process of reporting and tracking bugs by hand. It also eliminates the need to create a custom solution using general purpose database tools such as 4D™ or FileMaker Pro™.

Communicate TestTrack links engineers, testers, managers, even tech writers together so no one falls out of the loop. Team members are notified automatically when defects are assigned to them, guaranteeing communication and ensuring efficient work flow.

Stay up to Date TestTrack lets any authorized user look up the current state of any defect at any time.

Analyze TestTrack makes reporting easy—point, click, print and read. Customize reports to list what you want to see.

For a limited time, you can buy TestTrack for the introductory price of $99! Volume pricing, and site licenses are also available.

To order call 513-683-6456

Seapine Software, Inc. 1066 Seapine Ct. Maineville, OH 45039

sales@seapine.com
http://www.seapine.com

Seapine Software

production level. Leaving the space would have led to strange and unpleasant results.

By the way, there is a bug (or, say, a slight but unpleasant defect) in this HTML page; can you figure it out? Looking attentively at Figure 2 might help.

ALTERNATIVE SOLUTIONS, SHORTCOMINGS AND FUTURE WORKS

There are many unsatisfactory details in the solution we present here, either for the user experience or the developer's technical investment. We are working to fix or bypass some, but many remain beyond our reach, as we do not control the technology.

Filtering

Alternative forms (accessible from the page of Figure 1) allow some filtering on a French-English-Malay dictionary. As the kind of filtering is dependent on the selected dictionary, it was impossible to provide a single interface giving simultaneously the dictionary choice and the filtering options.

Diacritics

A common problem is encountered with diacritics. To properly produce HTML pages, it is compulsory to process each string of the dictionary item and to replace dubious characters by HTML codes. The little bug of Listing 10 is contained in "the apple of o's eye". The closing single smart quote character is not a valid one in HTML, resulting in poor formatting in Figure 2.

Speech

We are considering giving speech output to some information associated with an entry of our dictionary. On the Web server, a Text-To-Speech system could be installed and produce sound files to be downloaded by the client. But that's another story.

CONCLUSION

Technology integration is difficult for many reasons: you have to be reasonably knowledgeable in the different technologies you deal with, you have to compromise between many factors, and the whole thing is often hairy to debug.

As stated by Apple, AppleScript is the way to integrate technologies on Mac OS; but lots of people find it difficult to develop AppleScript-aware applications. With Macintosh Common Lisp, AppleScript awareness is given for free (the `do script` command). You can come up with a functional result quickly, satisfy your customers, and have a great tool to experiment with ideas and automated tasks.

ACKNOWLEDGMENTS

We would like to thank all the people who participated in the development of the tools mentioned in this paper: Apple and Digitool, of course, but also Jon Wiederspan for his CGI-aware script samples. Thank you, Bill, for the tremendous support to MCL and Wood. We would also not wish to forget all the people involved in our French-English-Malay Dictionary, the staff of the UTMK laboratory in Penang, the French Embassy in Malaysia, the DBDP in Kuala Lumpur, and the GETA in Grenoble, France.

[If you're curious about the authors and their work (and surroundings), start with the following Web pages, which I uncovered by guesswork (no word-searching necessary). You may need to brush up on your French first! – man]

http://clips.imag.fr/
http://www.imag.fr/IMAG/IMAG.html
http://www-clips.imag.fr/geta/
http://www-clips.imag.fr/geta/gilles.serasset/
http://www.usm.my/
http://161.142.8.195/wSambre-us/sambre-welcome-us.html
http://161.142.8.195/wSambre-us/Staff/ML-us.html

Visit MacTech Magazine's Web site!
http://www.mactech.com

SYMANTEC TOP 10

By Craig Conner

This monthly column, written by Symantec's Technical Support Engineers, is intended to give our readers technical information on using Symantec products.

This month we take a look at some new technologies, and then look at SPM. We also visit a few interesting topics from the newsgroups.

Q: I would like to do development of QuickDraw 3D applications using SPM, but I am getting "unrecognized pragma" warnings after patching the QD3D headers and samples.

A: The latest versions of the QuickDraw 3D headers (from the April '96 SDK) do not need any changes; however, they do include a pragma option, `!align_arrays`, which is not supported (and not necessary) on a PPC machine. Since it is just a warning, you can safely ignore it.

Also, to compile and debug the sample projects, you will need to:

1. open and update the project files;
2. enable Exception Handling in the PowerPC translator;
3. enable Background Events; and
4. replace the `.xcoff` libraries with their non-`.xcoff` versions using the **Script** menu item **Replace .xcoff libs**.

Q: Now that SPM supports 68K, can I create 68K code resources?

A: Yes. You will need to specify the correct link options via the Link Options, and you will need to specify the resource type and ID in the Command Line Linker Options page in the Additional Command Line Options.edit pane with `-rt type=ID`. Also, the SPM adds a `'SIZE'` resource to your target, which can be removed with ResEdit. Contact Symantec Technical Support for a project model to ease set-up.

Q: Can I use SPM to build Apple Shared Library Manager libraries?

A: Yes. There is a project model for building shared libraries on our ftp site at ftp://ftp.symantec.com.

Q: I would like to be able to write 68K assembly code, but the 8.1 C translator does not allow that. Is there a way to use 68K assembly code in SPM?

A: Yes. In the `Goodies:Tech Support Goodies` folder are 68K Assembly and 68K Pascal translators. These two translators allow you to compile 68K code via ToolServer, but provide a consistent user interface within the Symantec Project Manager. These two translators also support the new ability of SPM to combine 68K and PPC source and libraries within the same project, to create a fat application using a single project file.

Q: I want to use MrC++ to build a VA application, but if I specify MrC and MrC++ as the translators, I get a bunch of undefined Bedrock symbols.

A: MrC and MrC++ do not currently support the native exception handling of the Symantec C and C++ translators, but by replacing `BRLib.o` and `PPCCPlusLib TCL.o` with `BrLib (non-native eh).o` and `PPCCPlusLib TCL_BRLibeh.o` [*I suspect he means* `PPCCPlusLib TCL_BELeh.o` *here – man*], you can use MrC and MrC++ to generate your VA application.

Q: I am getting this link error when building an application in the SPM:

```
"Toolserver 's must occur in pairs."
```

What causes it?

POSITIONS WANTED

Available Immediately 1-800-736-3577

Expert 4D Programmers Less than 50¢ per hour!

More than 1,000 hours of development went into 4D Toolkit 2.0. With a one time price of $395.00*, it's like hiring a crack team of 4D programmers at less than 50¢ per hour. Call 1-800-736-3577 to order your copy, or to receive a free demo.

4D TOOL**KIT**™

OPtions Computer Consulting
228 Bleecker Street #19
New York, NY 10014
TEL: 212-645-3577
FAX: 212-633-0336

* *upgrade pricing available*

A: Most likely, you have a single-quote character in a folder located in the path to your project folder, something like `Dave's Folder`. ToolServer makes special use of single-quote characters in pathnames. Removing the apostrophe will fix the problem but make your filenames sound ungrammatical. So much for good English.

Q: When trying to create a PCI driver, I get these link errors:

```
Multiply Defined Symbol:
   BlockMove (DriverServicesLib, InterfaceLib)
Multiply DefinedSymbol:
   BlockMoveData (DriverServicesLib, InterfaceLib)
```

The libraries involved are:

`DriverServicesLib 1.0.2` from the April 96 SDK 2 disk
`InterfaceLib 1.1` from our Release 5 CD

How do I avoid this?

A: This is a bug in the `DriverServicesLib`. These symbols should not have been redefined in this library. Hopefully, by the time you read this Apple will have released a newer version.

Q: And now a Java question. When compiling my Java project I am getting the compile error:

```
Error: File Foo.class does not contain Foo as expected,
but    java.somepackage.Foo.  Please remove the file.
```

What gives?

A: The project is looking for something in the `java.somepackage.Foo` package, but can not find it. Most likely you forgot to `import` it. Add an `import java.somepackage.Foo` line and it should fix it.

Q: Is there a TCL archive somewhere?

A: You can now find indices at:
http://rhino.harvard.edu/dan/TCLArchive.html
and the files themselves at:
ftp://rhino.harvard.edu/pub/dan/TCL/

If you have classes you want to add, upload them to:
ftp://rhino.harvard.edu/incoming/
and send Dan Crevier at dan@rhino.harvard.edu a message about it. By the way, thanks, Dan.

Q: I am trying to port a PowerPlant application to Symantec C++, but I get the following error:

```
File "PPobClasses.cp"; Line 57
Error:   cannot implicitly convert
from: LButton *(*"C++")(LStream *)
to  : void *(*"C++")(LStream *)
```

on the line:

```
URegistrar::RegisterClass(LButton::class_ID,
   LButton::CreateButtonStream);
```

Why?

A: The second parameter of `RegisterClass()` is a `ClassCreatorFunc`, defined as:

```
typedef void* (*ClassCreatorFunc)(LStream *);
```

The problem is that each object's creator function (as for `LButton` above) returns a pointer to a class, which violates contravariance rules of C++.

Thanks to multiple people here in tech support, who each individually probably wrote more for this article than I did.

Visit MacTech Magazine's Web site!
http://www.mactech.com

You Can't Wait Forever.

Let's Face It.

As a professional developer or webmaster, you have to think of yourself first. You have deadlines to meet, products to deliver, and people depending on you.

A Tool in Hand...

Your tools are your livelihood. You need up-to-date technologies and timely support. You can't afford to wait for vaporware to materialize or for your tools vendor to leisurely add features that you need today.

When Hell Freezes Over?

When was your text editor last updated? Six months? A year? Five years? And when you contact the vendor that produced it, do they tell you how they're working on the next great version, how wonderful it's going to be, and how it will ship "Real Soon Now"? Or worse yet, do you get a disconnected phone number?

That's where we come in.

At Bare Bones Software, we deliver the best tools here and now. We have a distinguished tradition of continually enhancing the functionality, performance, and design of our products to meet our customers' needs. We don't allow our products to languish until we feel like doing something about it. We won't hype vaporware as the solution to today's problems. And we haven't drifted in and out business on a whim. **We're here to support you for the long haul.**

What About the Software?

You don't have to wait for the software, either. We're constantly refining our techniques and technology to bring you the fastest and most efficient tools. As the result of our efforts, BBEdit has repeatedly raised the standard for Macintosh text editing. (In fact, we've done so many cool things, several of our competitors have tried to copy us. Imitation is the sincerest form of flattery, after all. We're *very* flattered.)

Why Believe Us?

Since its commercial debut in 1993, BBEdit has set the pace for capability, user-interface, functionality, and customer support that others have yet to match. BBEdit was the first text editor to support ToolServer, THINK Reference, Toolbox Assistant, THINK C/Symantec C++, Internet Config, Quickdraw GX printing, PowerTalk, PowerPC acceleration, AppleScript, Apple Guide, and Macintosh Drag and Drop. We're still setting the pace for others to follow, as the only Macintosh text editor to integrate directly with Metrowerks CodeWarrior, through the use of the ClickWarrior extension.

See for Yourself.

We've painted a great picture, but don't take our word for it. Visit our Web site and see for yourself. You'll find lots of things to like.

BBEdit. Because you can't wait forever.

B
Bare Bones
Software, Inc.

Bare Bones Software, Inc.
P.O. Box 1048
Bedford, MA 01730-1048 USA

Voice: (508) 651-3561
Fax: (508) 651-7584
E-mail: bbsw@netcom.com
Web: http://www.tiac.net/biz/bbsw/

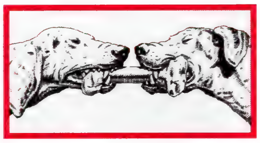
BMUG
Choice Product
FALL 1995

Accelerated for Power Macintosh

Mac™OS

Our lawyers insist we say: BBEdit and our spiffy logo are trademarks of Bare Bones Software, Inc. Mac and the MacOS logo are trademarks of Apple Computer, Inc., used under license. PowerPC is a trademark of International Business Machines Corporation. All other trademarks and registered trademarks are properties of their respective holders. Bare Bones Software, Inc. products are cruelty free (except for a few beta testers). © 1996 Bare Bones Software, Inc.

PROGRAMMER'S CHALLENGE



By Bob Boonstra

CONNECT IV

This month, your Challenge is to write code for a well-known game that will compete against other Challenge entries, as well as against the clock. The game board consists of an `N×M` array into which two players alternate inserting pieces. Pieces are inserted into the top of a column in the vertically oriented board, so that they drop into the lowest unoccupied cell of that column. The objective is to be the first player to arrange 4 or more pieces into a vertical, horizontal, or diagonal line.

The winner of this Challenge will be determined by a round-robin tournament in which each entry competes against each other entry an even number of times, half of the time playing with the first move, and half playing with the second move. (If the number of entries makes a full round-robin tournament impractical, I'll use another fair tournament scheme.) Each tournament victory will earn two points for the winning entry, and each tie earns one point. The entry with the greatest number of tournament points wins the Challenge. Ties will be broken by execution time, code size, etc.

The prototype for the code you should write is:

```
long /*yourMove*/ ConnectMove (
  long numCols,          /* number of columns in the game board */
  long numRows,          /* number of rows in the game board */
  void *privStorage,     /* preallocated storage for your use */
  long prevMove,     /* column where opponent last moved, 0 .. numCols-1 */
  Boolean newGame,       /* TRUE if this is your first move in a new game */
  Boolean *victory       /* set to TRUE if you claim victory */
                         /*  this move, otherwise set to FALSE */
);
```

For your first move in a new game, your `ConnectMove` code will be called with `newGame` set to `TRUE`. If your opponent made the first move in this game, `prevMove` will be set to the (origin zero) column that the opponent selected as the first move. If you are to make the first move, `prevMove` will be set to `-1`. In the call for the first move of a new game, you will also be provided with a pointer `privStorage` to 1MB of memory, already initialized to zero, for your use during that game. Finally, the initial call will provide the number of columns (`numCols`) and rows (`numRows`) in the board for that game. The number of rows and columns will each be larger than 6 and less than 64.

THE NEW DEADLINE

The Challenge is due on the FIRST of the month, not the tenth, as has been the case since the column started. To get the Challenge problem in time, you should be subscribed to the Programmer's Challenge Announcement List.

Subscribe by sending an email to "macjordomo@listmail.xplain.com", with the MESSAGE "sub challenge-a Your Name".

There's also an open discussion list to discuss Challenges, review solutions, and talk about efficient computing. You subscribe to this list by sending an email to "macjordomo@listmail.xplain.com", with the MESSAGE "sub challenge-d Your Name".

And finally, there's an ftp site for members of the Challenge lists. Its URL is: ftp://ftp.mactech.com/mactech/CHALLENGE.

People have used the ftp site to trade sample data and upload alternative solutions.

THE RULES

Here's how it works: Each month we present a new programming challenge. First, write some code that solves the challenge. Second, optimize your code (a lot). Then, submit your solution to *MacTech Magazine*. We choose a winner based on code correctness, speed, size, and elegance (in that order of importance) as well as the submission date. In the event of multiple equally desirable solutions, we'll choose one winner (with honorable mention, but no prize, given to the runner up). The prize for each month's best solution is a $100 credit for Developer Depot™ and a limited-edition "The Winner! MacTech Programmers Challenge" T-shirt (not available in stores anywhere).

Unless stated otherwise in the problem statement, the following rules apply: All solutions must be in ANSI compatible C. Use only pure C code. We disqualify entries with any assembly in them (except for challenges specifically stating otherwise). You may call any Macintosh Toolbox routine (e.g., it doesn't matter if you use NewPtr instead of malloc). We test entries with compiler options set to disable FPU use (for 680x0 code) and to enable all available speed optimizations. The compiler to be used and the target instruction set (680x0 or PowerPC) will be stated in the problem. **Limit your code to 60 characters per line;** this helps with e-mail gateways and page layout.

We publish the solution and winners for each month's Programmer's Challenge two months later. All submissions must be ***received by*** the 1st day of the month printed on the front cover of this issue.

You can get a head start on the Challenge by reading the Programmer's Challenge mailing list. It will be posted to the list on or before the 12th of the preceding month. To join, send an email to macjordomo@listmail.xplain.com with the message "sub challenge-A YourName'.

Mark solutions "Attn: Programmer's Challenge Solution" and send it by e-mail to one of the Programmer's Challenge addresses in the "How to Communicate With Us" section on page 2 of this issue. Include the solution, all related files, and your contact info.

MacTech Magazine reserves the right to publish any solution entered in the Programmer's Challenge. Authors grant *MacTech Magazine* the exclusive right to publish entries without limitation upon submission of each entry. Authors retain copyrights for the code.

Alternating calls will be made to your `ConnectMove` routine and the opposing `ConnectMove` routine. For the second and subsequent moves in each game, `newGame` will be set to `FALSE`, and the values of `numCols`, `numRows`, and `privStorage` will be the same as in the initial call. Each time `ConnectMove` is called, you should evaluate the move made by your opponent (provided in `prevMove`) and return the column into which you make your move as the function result. If you believe you have won the game, you should set `*victory` to `TRUE`; otherwise you should set `*victory` to `FALSE`. Falsely claiming victory or making an illegal move, such as trying to insert a game piece into a column that is already filled, will result in forfeiting that game.

For this Challenge, you may dynamically allocate additional storage beyond that provided by `privStorage`, provided you deallocate it before returning after each move. You may not use any static storage beyond that provided by `privStorage`.

To conduct the tournament, the code for multiple entries will be compiled into a single application. In case you are wondering, I will append a unique integer to the name of each routine before compiling so that each `ConnectMove` routine will be named uniquely.

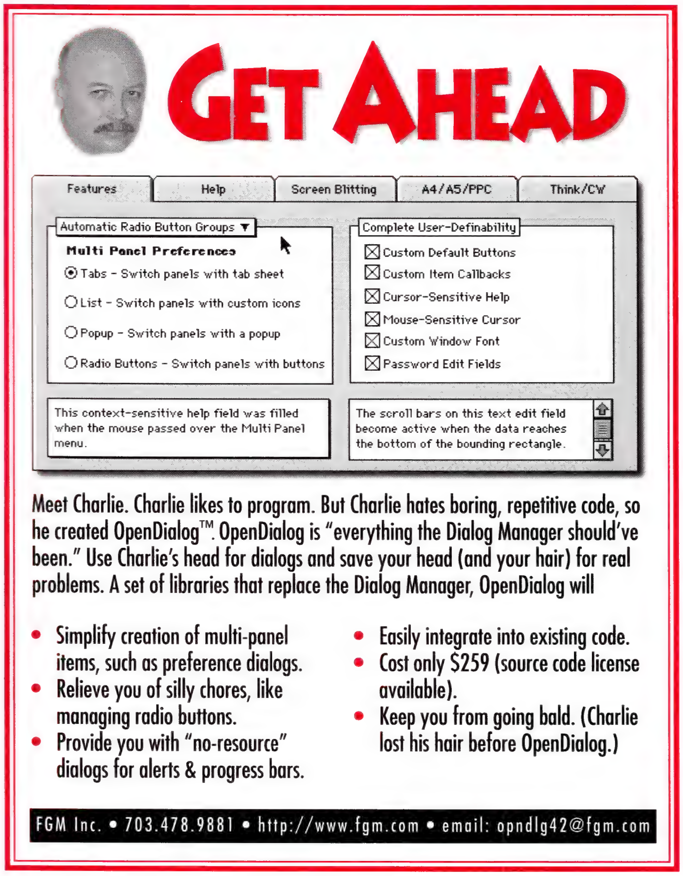
This will be a native PowerPC Challenge, scored using the Symantec environment. Solutions may be coded in C or C++.

Two Months Ago Winner

Perhaps the earlier Challenge deadline caught people a little off guard this month. Whatever the reason, only three people chose to enter the Edge Detector Challenge this month. Congratulations to **Ernst Munter** (Kanata, ON, Canada) for submitting the fastest entry to the Edge Detector Challenge. This is Ernst's third win in the last four contests, and it brings him within 2 points of the lead in our Top Contestants point total.

Contestants were required to process PixMap images in 8-bit, 16-bit, or 32-bit color and identify pixels where the color difference with adjacent pixels exceeded a specified threshold in the specified color component (or combination of components). The second place solution by **Miguel Cruz Picao** was actually faster than the winning solution in two of the three color depth cases, but was slower in the third case and slower overall. Miguel elected to implement entirely separate code for each of the three color depth cases, while Ernst sacrificed some processing time for smaller code size by testing for color depth while processing each line of the image, using common code for the rest of the three cases. On the other hand, Ernst used the `EdgeType` provided as a parameter to select, outside of any loops, the edge detection routine to use for the entire image, while Miguel's code tested the `EdgeType` for each pixel. Ernst elected to make four passes through each pair of adjacent image lines, comparing pixels that were adjacent horizontally, diagonally in each direction, and vertically, allowing compact and efficient comparison loops. The overall result was that the winning solution was some 25% faster than the second-place solution in the 32-bit case, which was enough to make it about 5% faster overall.

Table 1 summarizes the results for each entry, including code size, data size, execution time for each of the three color depth cases, and the overall score based on total execution time. Numbers in parentheses after a person's name indicate that person's cumulative point total for all previous Challenges, not including this one.



GET AHEAD
Features
Help
Screen Blitting
A4/A5/PPC
Think/CW
Automatic Radio Button Groups ▼
Multi Panel Preferences
● Tabs - Switch panels with tab sheet
○ List - Switch panels with custom icons
○ Popup - Switch panels with a popup
○ Radio Buttons - Switch panels with buttons
Complete User-Definability
☒ Custom Default Buttons
☒ Custom Item Callbacks
☒ Cursor-Sensitive Help
☒ Mouse-Sensitive Cursor
☒ Custom Window Font
☒ Password Edit Fields
This context-sensitive help field was filled when the mouse passed over the Multi Panel menu.
The scroll bars on this text edit field become active when the data reaches the bottom of the bounding rectangle.

Meet Charlie. Charlie likes to program. But Charlie hates boring, repetitive code, so he created OpenDialog™. OpenDialog is "everything the Dialog Manager should've been." Use Charlie's head for dialogs and save your head (and your hair) for real problems. A set of libraries that replace the Dialog Manager, OpenDialog will

- Simplify creation of multi-panel items, such as preference dialogs.
- Relieve you of silly chores, like managing radio buttons.
- Provide you with "no-resource" dialogs for alerts & progress bars.
- Easily integrate into existing code.
- Cost only $259 (source code license available).
- Keep you from going bald. (Charlie lost his hair before OpenDialog.)

FGM Inc. • 703.478.9881 • http://www.fgm.com • email: opndlg42@fgm.com

Name	Code	Data	Time (32)	Time (16)	Time (8)	Score
Ernst Munter (154)	1492	116	833	912	723	2467
Miguel Cruz Picao (11)	3420	20	1120	899	582	2600
Greg Cooper	1872	28	1830	1902	1802	5534

Table 1. Results for each entry

Rank Name	Points	Rank Name	Points
1. [Name deleted]	176	11. Mallett, Jeff	44
2. Munter, Ernst	174	12. Kasparian, Raffi	42
3. Gregg, Xan	92	13. Vineyard, Jeremy	42
4. Karsh, Bill	90	14. Lengyel, Eric	40
5. Larsson, Gustav	87	15. Darrah, Dave	31
6. Stenger, Allen	65	16. Brown, Jorg	30
7. Cutts, Kevin	57	17. Lewis, Peter	30
8. Riha, Stepan	51	18. Landry, Larry	29
9. Goebel, James	49	19. Beith, Gary	24
10. Nepsund, Ronald	47	20. Elwertowski, Tom	24

Table 2. Top 20 contestants

Wow!
Oh!
Yeah!
Oo!
Right On!

Recent Reactions To New QUED/M 3.0

Why does the new QUED/M 3.0 text editor elicit such responses? Because it's so fun to use, that's why. From the moment you see its inviting interface and access its numerous time-saving features, you'll likely have a few happy responses of your own to give.

QUED/M 3.0 makes editing fun by making it easy. Some of the many features you'll appreciate right away:

◆ Macros with a powerful new Programming Dialect provide built-in, Basic-like commands
◆ Integrated support for THINK C, CodeWarrior, MPW 411 and ToolBox Assistant lets you use QUED/M's advanced editor for all your editing
◆ Dynamic coloring of C/C++ keywords and comments color codes your work
◆ Unsurpassed editing capabilities, including System 7.5 Drag and Drop, noncontiguous selection, unlimited undos, GREP find/replace, Frontier™ and AppleScript support, file comparison, text folding, and more!
◆ Power PC code speeds up editing, searching and macros
◆ HTML macros included for creating Web pages
◆ FREE Apprentice 4 gives you over 600 MB of working source code to include in your own programs. MailKeeper e-mail organizer is included too!
◆ All this and more, for less than you'd pay for other text editors!

Start Having Fun. Order Qued/M 3.0 Today. Get your copy for just $69 at CYBERIAN Outpost **800-856-9800,** http://www.cybout.com/ • info@cybout.com. Or for more information, call 800-281-0101, or 619-481-4366 for Info-by-Fax. http://www.nisus-soft.com

©1996 Nisus Software Inc. QUED/M is a trademark, and Nisus is a registered trademark of Nisus Software Inc. All other product names are the property of their respective owners. Nisus main line: 619-481-1477

NISUS
Software Inc.

TOP 20 CONTESTANTS OF ALL TIME

Table 2 shows the Top 20 Contestants for the Programmer's Challenges to date The numbers below include points awarded for this month's entrants.

There are three ways to earn points: (1) scoring in the top 5 of any Challenge, (2) being the first person to find a bug in a published winning solution or, (3) being the first person to suggest a Challenge that I use. The points you can win are:

1st place	20 points
2nd place	10 points
3rd place	7 points
4th place	4 points
5th place	2 points
finding bug	2 points
suggesting Challenge	2 points

Here is Ernst's winning solution:

EDGEDETECTOR.C

Copyright 1996, Ernst Munter

/*
Problem Statement
—————————-
Given a Pixmap image, create a Bitmap which shows the edges in the image.

Pixmaps may have 8, 16, or 32 bit colors. Parameters control which colors should contribute to the edge detection, and a threshold for the color space difference (an RMS value) to be considered an edge.

All 16 bits of the individual color values must be considered, which can result in 32-bit overflow when squares are added.

Solution
————

The general idea is to process 2-row strips of the image through an intermediate ColorSpec stage in which the "value" fields are used to mark edge pixels.

Step1:

We allocate a small amount of memory, sufficient to hold two rows of pixels in ColorSpec format (8 bytes per pixel). Pixels from each row in the Pixmap are converted to ColorSpec format (RGB + "value").

Step2:

The two lines are scanned with 4 different patterns to determine edges: given 2 rows with pixels as shown

```
  A B C D E F G H ...
  a b c d e f g h ...
```

We compare pixel pairs in four separate runs of EdgePairs():

```
  a-b, b-c, c-d, d-e ...  (WEST - EAST)
  A-b, B-c, C-d, D-e ...  (NORTH-WEST - SOUTH-EAST)
  B-a, C-b, D-c, E-d ...  (NORTH-EAST - SOUTH-WEST)
  A-a, B-b, C-c, D-d ...  (NORTH - SOUTH)
```

This procedure is repeated for rows 0 and 1, then 1 and 2, 2 and 3, and so on. (Plus an West-East run for row 0).

During each run, the "value" field of every pair of pixels found to be part of an edge is set to -1 (0xFFFF). All other values remain at 0.

Step 3:

Before shifting down to the next row, the ColorSpecs of the upper row are scanned to construct a row of the Bitmap.

Optimizations
——————-

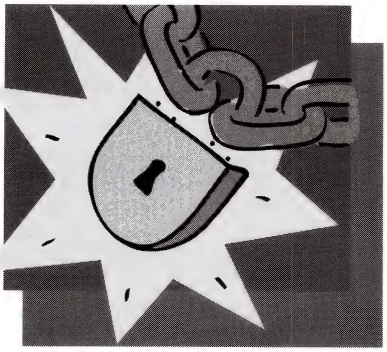
The combinations of pixel size and edge types are dealt with as follows:

dtF The Relational Database System

dtF v1.6 including numerous SQL enhancements and ODBC driver available!

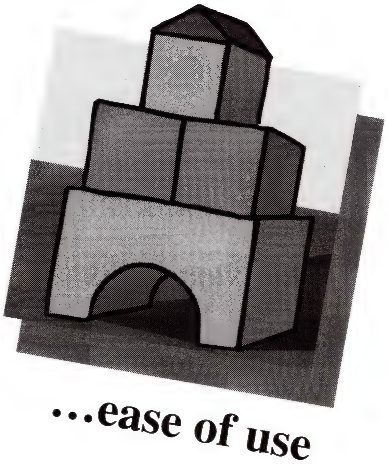
When it comes to performance **dtF** is in a class all its own. **dtF** utilizes a proprietary query optimization and caching scheme to obtain unparalleled performance. If you are in need of a quantum leap in performance, use the **dtF** native PowerPC stand-alone or client/server version.

...performance



Full transaction control and automatic level-3 error recovery guarantee maximum data protection even after sudden system crashes. **dtF** databases are compressed and encrypted to protect against all unauthorized access, even disk editors. Implement sophisticated table and operation level security with **dtF**'s password features.

...data security



With **dtF**'s unique direct integration technology, the **dtF** database system is fully contained in your application, enabling you to build double-clickable database applications. Never worry about missing or conflicting INITs or drivers. **dtF**'s native API is compact, easy-to-use and integrates seamlessly with all major development environments on the Macintosh. **dtF**'s high performance SQL, integrated data dictionary, security features, automatic index selection, query optimization and error recovery allow you to concentrate on creating great database applications instead of messing around with the internals of the database system.

...ease of use

dtF is available for Macintosh System 7.x (68K and native PowerPC). **dtF** supports MPW C/C++, Symantec C/C++, Metrowerks CodeWarrior (all compilers 68K and native PPC). **dtF** is fully OpenDoc™ compatible. Separate versions for use with HyperCard 2.x, SuperCard 2.x, Smalltalk-Agents and Pictorius Peregrine are provided. AppleScript interface via DataScript™ for **dtF** from General Knowledge. **dtF** supports cross-platform development on Windows 3.11, Windows 95 and OS/2.

Stand-alone applications built with dtF are royalty-free!

dtF Americas, Inc.
19672 Stevens Creek Blvd.,
Suite 128
Cupertino, CA 95014
USA

Phone: (800) DTF-1790
Fax: (510) 828-8755
AppleLink: DTF.AMERICA
Internet: dtF.america@applelink.apple.com

dtF The Relational Database System

Power Mac and Macintosh Developers:

FIND OUT FAST WHAT'S GOING ON IN MEMORY

THE MEMORY MINE™

- See memory allocation in any open heap at a glance.
- Easily spot memory leaks.
- Flags heap corruption when it happens.
- Works with source level debugger to let you find memory problems fast.
- Stress applications on the fly with Purge, Compact, and Zap.
- Allocate memory at will for precise stress testing.
- Log heap data - easily document heap status over time.
- No need for source code: nothing inserted in code; no patches to the system.
- Works with 24-bit, 32-bit, and modern memory managers.

For Macintoshes with 68020 or better. Requires System 7.0 or later.

only $99 US
Order now from Adianta, Inc.
Phone: (415)781-8052 • FAX: (415)781-8053
AppleLink:ADIANTA • AOL:Adianta • Internet:adianta@aol.com
For VISA, MC, or American Express orders by mail, fax, or Applelink, please include name, address, card number, expiration date, and phone number or email address.
Also available through the MacTech Mail Order Store and APDA

for more information contact

Adianta, Inc. • 582 Market St #911 • San Francisco, CA 94104

All pixel types are normalized to a single ColorSpec format by the GetLine procedure.

Different edge types call for different series of tests. Specifically, 1, 2, or 3 colors can be optimized separately. On of a set of 7 separate short functions is selected at runtime. This requires the edge type to be evaluated only once, to select the EdgeXXX procedure that will be used many times as the image is scanned from top to bottom.

Comparing an RMS (root of sum of squares) value with a threshold is equivalent to comparing the sum of squares with the square of the threshold value. This avoids the need to compute square roots.

In the case of a single color edge type, the square of the difference would not need to be formed. A simple comparison of the threshold with the absolute difference would suffice. However, the runtime difference between a multiply and computing the absolute value would depend on CPU type etc, and would tend to be small. I opted for uniformity, and square all differences.

The dynamically allocated storage should typically fit inside the CPU cache, and not slow things down as it is read and written continuously.

Assumptions
——————-
Bitmap is large enough to hold the result. eType is within the range of redOnly to redGreenAndBlue, otherwise the call will certainly crash.

Amount of static memory = 8 pointers (32 bytes)
Amount of dynamic memory required is for 2 rows of ColorSpec records (a 1024 pix wide image -> 16K of storage).

```
*/

#include <QDOffscreen.h>
// Contains PixMap, BitMap and EdgeType defs

#include <stdlib.h>
```

Definitions

```
typedef enum
{
   redOnly = 1,
   greenOnly,
   redAndGreen,
   blueOnly,
   redAndBlue,
   greenAndBlue,
   redGreenAndBlue
} EdgeType;

typedef void EdgeProc(
   ColorSpec*      P,
   ColorSpec*      Q,
   int             size,
   unsigned long   TH);
```

Function Prototypes

```
void EdgeDetect(
   PixMapHandle    pMapH,
   BitMap          *bMap,
   unsigned short  threshold,
   EdgeType        eType);

void GetlLine(
   ColorSpec*      line,
   void*           pix,
   int             size,
   ColorSpec*      CT,
   int             pixelSize);

void EdgeRed(
   ColorSpec*      P,
   ColorSpec*      Q,
   int             size,
   unsigned long   TH);

void EdgeGreen(
   ColorSpec*      P,
   ColorSpec*      Q,
   int             size,
   unsigned long   TH);

void EdgeRedGreen(
   ColorSpec*      P,
   ColorSpec*      Q,
   int             size,
   unsigned long   TH);

void EdgeBlue(
   ColorSpec*      P,
   ColorSpec*      Q,
   int             size,
   unsigned long   TH);

void EdgeRedBlue(
   ColorSpec*      P,
   ColorSpec*      Q,
   int             size,
   unsigned long   TH);

void EdgeGreenBlue(
   ColorSpec*      P,
   ColorSpec*      Q,
   int             size,
   unsigned long   TH);

void EdgeRedGreenBlue(
   ColorSpec*      P,
   ColorSpec*      Q,
   int             size,
   unsigned long   TH);

void Copy2Bitmap(
   ColorSpec*      P,
   unsigned char*  bits,
   int             size);
```

WHY NOT USE THE BEST?

Companies like Adobe, Claris, Symantec, Metrowerks, Netscape and hundreds of others recognize the unsurpassed quality of Developer VISE and have switched to MindVision for their installation needs. In fact, every day MindVision sells more installers than all other competitors combined. Don't take our word for it. Ask any of our hundreds of satisfied customers why they switched from the competition to MindVision's Developer VISE.

We didn't stop there.
Announcing...

INSTALLER VISE 4.0

The New Industry Standard for Software Installers.

Find, move, rename, or delete any file. Create hierarchical packages. Build one installer for multiple languages. Build installers with unparalleled ease. Attach AppleScripts to your installation. New archive features allow you to build CD-ROM installers quickly and easily. It may not slice and dice, but in addition to installing it also removes. And much, much more.

And Introducing...

UPDATER VISE 1.1

The Most Advanced Updater Available.

Updater VISE is the only product that can update multiple versions of your 68K, PowerPC, and Fat application from ONE updater. Updaters are extremely compact and totally secure, allowing for easy online distribution. And, combined with Installer VISE you can update any number of different files giving you a total software delivery solution.

Maybe it's time to stop and take a look at MindVision's software delivery solutions.
Visit our web site for full information,
including fully-functional demos.

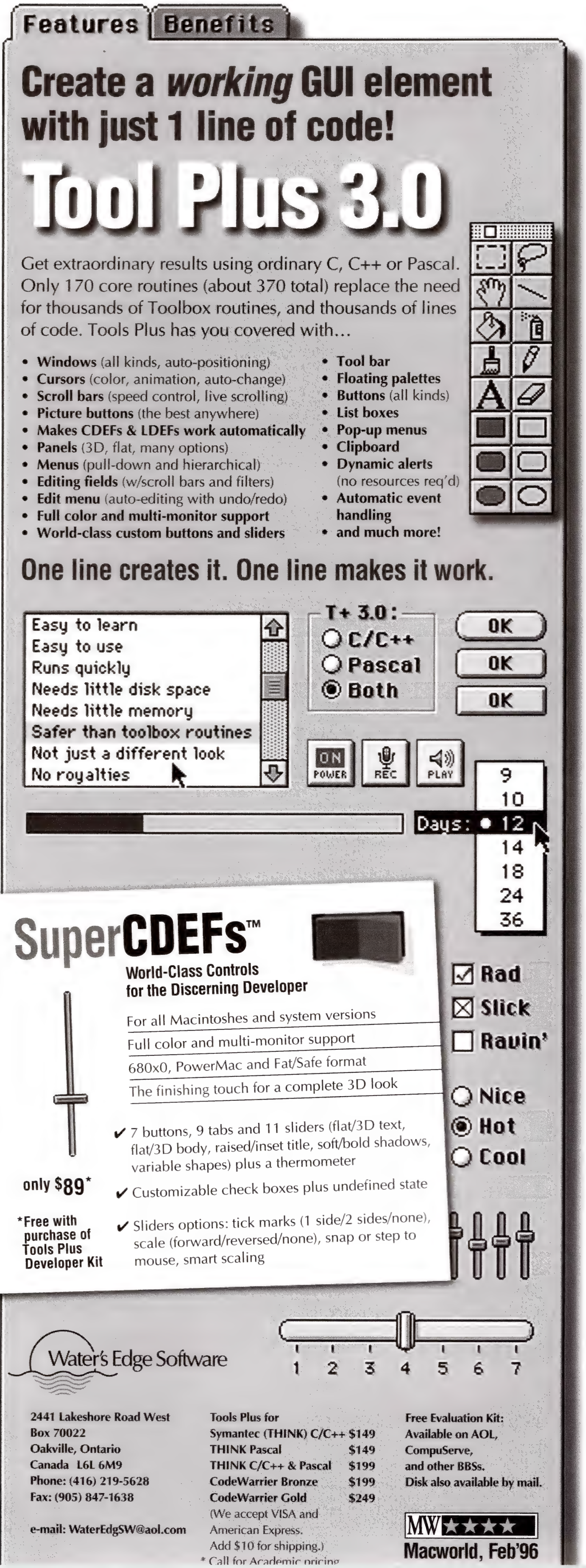
MindVision
Software

www.mindvision.com

Email: info@mindvision.com • Voice: (402) 477-3269 • Fax: (402) 477-1395

VISE
SOFTWARE
DELIVERY
SYSTEM

© 1996 MindVision Software. All Rights Reserved. Developer VISE, Installer VISE, and Updater VISE are trademarks of MindVision Software.



Features | Benefits

Create a *working* GUI element with just 1 line of code!

Tool Plus 3.0

Get extraordinary results using ordinary C, C++ or Pascal. Only 170 core routines (about 370 total) replace the need for thousands of Toolbox routines, and thousands of lines of code. Tools Plus has you covered with…

- **Windows** (all kinds, auto-positioning)
- **Cursors** (color, animation, auto-change)
- **Scroll bars** (speed control, live scrolling)
- **Picture buttons** (the best anywhere)
- **Makes CDEFs & LDEFs work automatically**
- **Panels** (3D, flat, many options)
- **Menus** (pull-down and hierarchical)
- **Editing fields** (w/scroll bars and filters)
- **Edit menu** (auto-editing with undo/redo)
- **Full color and multi-monitor support**
- **World-class custom buttons and sliders**
- **Tool bar**
- **Floating palettes**
- **Buttons** (all kinds)
- **List boxes**
- **Pop-up menus**
- **Clipboard**
- **Dynamic alerts** (no resources req'd)
- **Automatic event handling**
- **and much more!**

One line creates it. One line makes it work.

Easy to learn
Easy to use
Runs quickly
Needs little disk space
Needs little memory
Safer than toolbox routines
Not just a different look
No royalties

T+ 3.0:
○ C/C++
○ Pascal
● Both

OK
OK
OK

ON POWER
REC
PLAY

Days: 9 10 ● 12 14 18 24 36

☑ Rad
☒ Slick
☐ Ravin'

○ Nice
● Hot
○ Cool

1 2 3 4 5 6 7

SuperCDEFs™

World-Class Controls for the Discerning Developer

For all Macintoshes and system versions
Full color and multi-monitor support
680x0, PowerMac and Fat/Safe format
The finishing touch for a complete 3D look

✔ 7 buttons, 9 tabs and 11 sliders (flat/3D text, flat/3D body, raised/inset title, soft/bold shadows, variable shapes) plus a thermometer
✔ Customizable check boxes plus undefined state
✔ Sliders options: tick marks (1 side/2 sides/none), scale (forward/reversed/none), snap or step to mouse, smart scaling

only $89*

*Free with purchase of Tools Plus Developer Kit

Water's Edge Software

2441 Lakeshore Road West
Box 70022
Oakville, Ontario
Canada L6L 6M9
Phone: (416) 219-5628
Fax: (905) 847-1638

e-mail: WaterEdgSW@aol.com

Tools Plus for	
Symantec (THINK) C/C++	$149
THINK Pascal	$149
THINK C/C++ & Pascal	$199
CodeWarrier Bronze	$199
CodeWarrier Gold	$249

(We accept VISA and American Express. Add $10 for shipping.)
* Call for Academic pricing

Free Evaluation Kit: Available on AOL, CompuServe, and other BBSs. Disk also available by mail.

MW ★★★★★
Macworld, Feb'96

Static Allocations

```
static EdgeProc* EdgeProcedure[8] = {
  0,
  EdgeRed,
  EdgeGreen,
  EdgeRedGreen,
  EdgeBlue,
  EdgeRedBlue,
  EdgeGreenBlue,
  EdgeRedGreenBlue};
```

EdgeDetect

```
void EdgeDetect(
  PixMapHandle    pMapH,
  BitMap          *bMap,
  unsigned short  threshold,
  EdgeType        eType) {
PixMap*  pm=*pMapH;

unsigned char*  pix=(unsigned char*)GetPixBaseAddr(pMapH);
unsigned char*  bits=(unsigned char*)bMap->baseAddr;
unsigned long   TH=(unsigned long)threshold*threshold;
int    y;
int    pixelSize=pm->pixelSize;
int    pixRowBytes=pm->rowBytes & 0x3FFF;
int    pixInRow=pm->bounds.right-pm->bounds.left;
int    bitRowBytes=bMap->rowBytes;

ColorSpec* allocated;
ColorSpec* line0;
ColorSpec* line1;
ColorSpec* CTable=(*(pm->pmTable))->ctTable;
EdgeProc*  EP;

// Get memory for a 2-row strip of pixels:
  if (0==(allocated=(ColorSpec*)malloc(
  2*sizeof(ColorSpec)*pixInRow))) return;
  line0=allocated-1;   //(we always use pre-increment later)
  line1=line0+pixInRow;

// Determine EdgeProc to use:
  EP=EdgeProcedure[eType];

// Prepare first row:
  Get1Line(line0,pix,pixInRow,CTable,pixelSize);
  (*EP)(line0,line0+1,pixInRow-1,TH);    //W-E

// Process remaining rows:
  for (y=pm->bounds.top+1;y<pm->bounds.bottom;y++) {
    pix+=pixRowBytes;
    Get1Line(line1,pix,pixInRow,CTable,pixelSize);

    (*EP)(line1,  line1+1,pixInRow-1,TH);  //W-E
    (*EP)(line0,  line1+1,pixInRow-1,TH);  //NW-SE
    (*EP)(line0+1,line1,  pixInRow-1,TH);  //NE-SW
    (*EP)(line0,  line1,  pixInRow,TH);    //N-S

// Convert ColorSpec.values to Bitmap.bits:
    Copy2Bitmap(line0,bits,bitRowBytes);

// Swap line0 with line1:
    {ColorSpec* temp=line0;line0=line1;line1=temp;}
    bits+=bitRowBytes;
  }

// Convert the last row;
  Copy2Bitmap(line0,bits,bitRowBytes);

// Release dynamic memory:
  free(allocated);
}
```

Get1Line

```
/***************** Local Functions ***********************/
void Get1Line(
  ColorSpec* line,
  void*      pix,
  int        size,
  ColorSpec* CT,
  int    pixelSize){
```

```
int  i,r,g,b;
unsigned long   x;
  if (pixelSize==8) {
    unsigned char*  PIX =((unsigned char*)pix);
    for (i=0;i<size;i++) {
      x=*PIX++;
      line++;
      *((double*)(line))=*((double*)(CT+x));
      line->value=0;
    }
  } else if (pixelSize==16) {
    unsigned short* PIX =((unsigned short*)pix);
    for (i=0;i<size;i++) {
      x=*PIX++;
      line++;
      r=((x & 0x7C00) << 1)  | ((x & 0x7C00) >> 4) |
  ((x & 0x7C00) >> 9)  | ((x & 0x7C00) >> 14);
      g=((x & 0x03E0) << 6)  | ((x & 0x03E0) << 1) |
  ((x & 0x03E0) >> 4)  | ((x & 0x03E0) >> 9);
      b=((x & 0x001F) << 11) | ((x & 0x001F) << 6) |
  ((x & 0x001F) << 1)  | ((x & 0x001F) >> 4);
      line->value=0;
      line->rgb.red=  (unsigned short)r;
      line->rgb.green=(unsigned short)g;
      line->rgb.blue= (unsigned short)b;
    }
  } else {  //pixelSize not 8 or 16, so it must be 32
    unsigned long*  PIX =((unsigned long*)pix);
    for (i=0;i<size;i++) {
      x=*PIX++;
      line++;
      r=((x & 0xFF0000) >> 8) | ((x & 0xFF0000) >> 16);
      g= (x & 0x00FF00)       | ((x & 0x00FF00) >> 8);
      b= (x & 0x0000FF)       | ((x & 0x0000FF) << 8);
      line->value=0;
      line->rgb.red=  (unsigned short)r;
      line->rgb.green=(unsigned short)g;
      line->rgb.blue= (unsigned short)b;
    }
  }
}
```

Copy2Bitmap

```
void Copy2Bitmap(
  ColorSpec* P,
  unsigned char*  bits,
  int    size) {
int    i;
  P++;
  for (i=0;i<size;i++) {
    *bits = (unsigned char) (
      (P[0].value & 0x80) |
      (P[1].value & 0x40) |
      (P[2].value & 0x20) |
      (P[3].value & 0x10) |
      (P[4].value & 0x08) |
      (P[5].value & 0x04) |
      (P[6].value & 0x02) |
      (P[7].value & 0x01) );
    P+=8;
    bits++;
  }
}
```

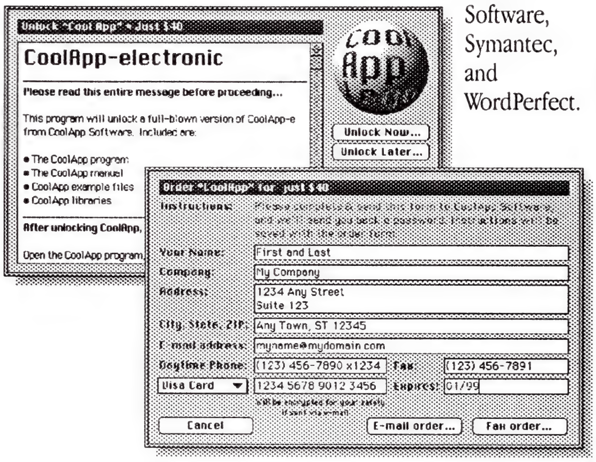
EdgeRedGreenBlue

```
void EdgeRedGreenBlue(
  ColorSpec*      P,
  ColorSpec*      Q,
  int             size,
  unsigned long   TH) {
int    i;
unsigned long   delta,acc;
  for (i=0;i<size;i++) {
    P++;Q++;
    acc=(unsigned long)P->rgb.red-Q->rgb.red;
    acc*=acc;
    delta=(unsigned long)P->rgb.green-Q->rgb.green;
    delta*=delta;
    acc+=delta;
    if (acc<delta) {P->value=Q->value=-1;continue;}
    delta=(unsigned long)P->rgb.blue-Q->rgb.blue;
```

SOLID SOLUTION FOR ELECTRONIC DISTRIBUTION

BroadCast™

BroadCast is a highly secure, patent-pending system for safely distributing "locked" software online, or on CD-ROM. This cutting-edge technology was developed by the same engineering team behind PatchWorks— our tool for creating online updaters that's become an industry standard used by clients like Apple Computer, Adobe Systems, MacroMind, Now Software, Symantec, and WordPerfect.



CoolApp-electronic

Please read this entire message before proceeding...

This program will unlock a full-blown version of CoolApp-e from CoolApp Software. Included are:

- The CoolApp program
- The CoolApp manual
- CoolApp example files
- CoolApp libraries

After unlocking CoolApp,

Open the CoolApp program,

Your Name: First and Last
Company: My Company
Address: 1234 Any Street
Suite 123
City, State, ZIP: Any Town, ST 12345
E-mail address: myname@mydomain.com
Daytime Phone: (123) 456-7890 x1234 Fax: (123) 456-7891
Visa Card 1234 5678 9012 3456 Expires: 01/99

BROADCAST IS EASY FOR YOU AND YOUR CUSTOMERS

In seconds, BroadCast securely locks your product into an Unlocker application that can be safely distributed to the public. To buy your product, customers simply run the Unlocker— then either call you, or complete an onscreen order form that's emailed or faxed to you with a unique "control number" (and encrypted credit card data).

Your staff processes the credit card, registers your new user, and provides your customer with a password based upon the control number. To unlock your software, the customer enters this password in the Unlocker.

BROADCAST IS INEXPENSIVE

We aren't resellers or distributors. Customers buy directly from you, and we take no percentage of your sales— just a small fee per transaction. In fact, BroadCast is distributed for free. We're betting you'll quickly use the free password points that come with BroadCast, and purchase more as your sales soar.

We'll even help attract customers by advertising your products for free on our "Software Unboxed" internet mall.

GET THE WHOLE STORY

We invite you to learn more by visiting our internet web site at "http://www.broadcastsoft.com", and download your free BroadCast Starter Kit, or email us: "info@broadcastsoft.com".

BroadCast

BroadCast Software Corp
East: 407-241-0308, Fax: 407-241-3195
West: 503-317-0429, Fax: 503-317-0430
Net: http://www.broadcastsoft.com
Email: info@broadcastsoft.com

Programmer Training

MACINTOSH SEMINARS & CONSULTING

Now **WindowsNT** cross-platform training

Richey Software Training provides professional, *customized* programming seminars and industry consulting.

Rich content & hands-on lab exercises shorten your learning curve. On-site training is convenient — your team eliminates expensive travel costs and consuming down-time.

“*Professional training & consultancy that really hits the mark.*”

MAC SEMINARS

- C & C++
- OOP
- PowerPC
- AppleScript
- Debugging

Call today to find out how Richey Software Training delivers professional seminars and consulting nationwide.

RICHEY SOFTWARE TRAINING

Training Mac Programmers Since 1986

707•869•2836

e-mail us at dave@richey.com

or check us out on the web at www.richey.com

P.O. BOX 1809, GUERNEVILLE, CA 95446-1809

© 1996 Richey Software Training. All trademarks or registered trademarks are the property of their respective owners. Specifications subject to change.

```
    delta*=delta;
    acc+=delta;
    if ((acc>=TH) || (acc<delta)) P->value=Q->value=-1;
  }
}
```

EdgeGreenBlue

```
void EdgeGreenBlue(
  ColorSpec*      P,
  ColorSpec*      Q,
  int             size,
  unsigned long   TH) {
int    i;
unsigned long  delta,acc;
  for (i=0;i<size;i++) {
    P++;Q++;
    acc=(unsigned long)P->rgb.green-Q->rgb.green;
    acc*=acc;
    delta=(unsigned long)P->rgb.blue-Q->rgb.blue;
    delta*=delta;
    acc+=delta;
    if ((acc>=TH) || (acc<delta)) P->value=Q->value=-1;
  }
}
```

EdgeRedBlue

```
void EdgeRedBlue(
  ColorSpec*      P,
  ColorSpec*      Q,
  int             size,
  unsigned long   TH) {
int    i;
unsigned long  delta,acc;
  for (i=0;i<size;i++) {
    P++;Q++;
    acc=(unsigned long)P->rgb.red-Q->rgb.red;
    acc*=acc;
    delta=(unsigned long)P->rgb.blue-Q->rgb.blue;
    delta*=delta;
    acc+=delta;
    if ((acc>=TH) || (acc<delta)) P->value=Q->value=-1;
  }
}
```

EdgeRedGreen

```
void EdgeRedGreen(
  ColorSpec*      P,
  ColorSpec*      Q,
  int             size,
  unsigned long   TH) {
int    i;
unsigned long  delta,acc;
   for (i=0;i<size;i++) {
    P++;Q++;
    acc=(unsigned long)P->rgb.red-Q->rgb.red;
    acc*=acc;
    delta=(unsigned long)P->rgb.green-Q->rgb.green;
    delta*=delta;
    acc+=delta;
    if ((acc>=TH) || (acc<delta)) P->value=Q->value=-1;
  }
}
```

EdgeBlue

```
void EdgeBlue(
  ColorSpec*      P,
  ColorSpec*      Q,
  int             size,
  unsigned long   TH) {
int    i;
unsigned long  acc;
  for (i=0;i<size;i++) {
    P++;Q++;
    acc=(unsigned long)P->rgb.blue-Q->rgb.blue;
    acc*=acc;
    if (acc>=TH) P->value=Q->value=-1;
  }
}
```

EdgeGreen

```
void EdgeGreen(
  ColorSpec*      P,
  ColorSpec*      Q,
  int             size,
  unsigned long   TH) {
int    i;
unsigned long  acc;
  for (i=0;i<size;i++) {
    P++;Q++;
    acc=(unsigned long)P->rgb.green-Q->rgb.green;
    acc*=acc;
    if (acc>=TH) P->value=Q->value=-1;;
  }
}
```

EdgeRed

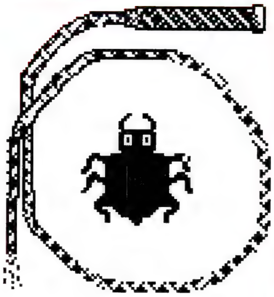
```
void EdgeRed(
  ColorSpec*      P,
  ColorSpec*      Q,
  int             size,
  unsigned long   TH) {
int    i;
unsigned long  acc;
  for (i=0;i<size;i++) {
    P++;Q++;
    acc=(unsigned long)P->rgb.red-Q->rgb.red;
    acc*=acc;
    if (acc>=TH) P->value=Q->value=-1;
  }
}
```

Gives you the Information to *Program your Best!*

Information

The Debugger V2 & MacNosy

by Steve Jasik



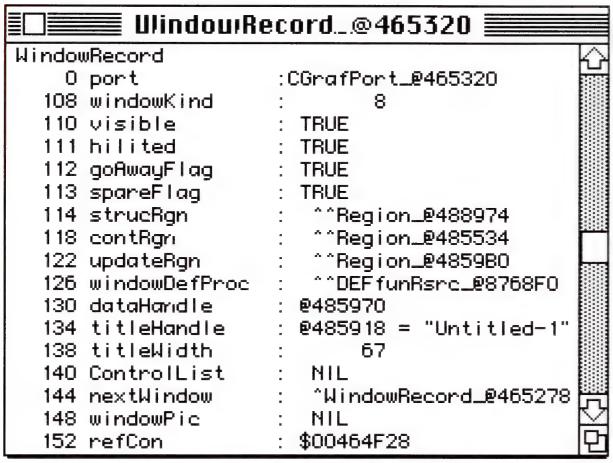
Control

For over 10 years Steve Jasik has brought Macintosh Developers the superior debugger. In addition to the software, Steve provides timely support to his customers. Inspect the feature list and see why so many developers use it. In addition to basic features such as stepping and breakpoints, the Jasik debugger has many advanced features such as heap display, a fast memory watch, Soft-MMU, jump tracing, ... to help you track down the elusive bugs that other debuggers won't help you find.

In additon, the package includes **MacNosy,** a global interactive disassembler that enables one to recover the source code of any Mac application or from the ROM.

When you compare features of the different debuggers, note that *only one* has all the below features to help you get your job done, and *only one* has MacNosy to help you debug any program in a full system environment symbolically!

"The Debugger" is the debugger of choice at: Adobe, Apple, Claris, Kodak, Metrowerks, etc.



```
WindowRecord_@465320
WindowRecord
   0 port           :CGrafPort_@465320
 108 windowKind     :        8
 110 visible        : TRUE
 111 hilited        : TRUE
 112 goAwayFlag     : TRUE
 113 spareFlag      : TRUE
 114 strucRgn       :  ^^Region_@488974
 118 contRgn        :  ^^Region_@485534
 122 updateRgn      :  ^^Region_@4859B0
 126 windowDefProc  :  ^^DEFfunRsrc_@8768F0
 130 dataHandle     : @485970
 134 titleHandle    : @485918 = "Untitled-1"
 138 titleWidth     :       67
 140 ControlList    :  NIL
 144 nextWindow     :  ^WindowRecord_@465278
 148 windowPic      :  NIL
 152 refCon         : $00464F28
```

An example of a structured data display window with hypertext links to substructures

Its Features Include:

- **Source level debugging for Metrowerks & MPW** compiled programs (C++, C, Pascal, Fortran, ...).
- **Symbolic Debugging** of *any* Macintosh program, ROM, Plug-In or *code resource* (DRVRs, XCMDs, INITs, PDEFs..), ASLM & **CFM LIbraries, OpenDoc parts**, GX Drivers, ...
- **Simultaneous Symbolic debugging of multiple "tasks"**
- **Object Inspector for MacApp 3 & PowerPlant programs**
- **Source level debugging of Symantec C™ projects**
- **Includes a program (CoverTest) to interactively do Code Coverage analysis for SQA testing, etc.**
- **Fast Software Watchpoint** to find clobbered variables.
- **Soft MMU** ("heap centric" bounds checking) for PowerPC programs to find out of bounds references.
- **Jump Trace** to find where a program takes a "wild jump"
- **Sophisticated error check algorithms** such as Trap Discipline (Argument Checking), Heap Scramble and Heap Check to detect errors before they become disasters
- **Structured display** of data (hypertext) with user definable structures while debugging
- **Conditional breakpoints** to filter redundant information
- **Continuous Animated Step Mode** to watch your program execute instruction by instruction
- **Detailed symbolic disassembly for both 680x0 and PowerPC** with symbol names, labels, cross ref maps, – make it possible to ferret out the secrets of the ROM, etc.
- **"Training Wheels"** for the PowerPC disassembler to help you learn the opcodes

The Debugger V2 & MacNosy: $350
Runs on all Macs. Call For Group prices or Updates.
Visa/MC Accepted.
Available from: Jasik, APDA or ComputerWare (800-326-0092).

Jasik Designs • 343 Trenton Way, Menlo Park, California 94025 • (415) 322-1386
URL: http://www.jasik.com e-mail: macnosy@jasik.com

By Mike Scanlin

Code Fragments Explained

A shared library of procedure pointers for the resourceful programmer

**A Fragment of Your Imagination:
Code Fragments and Code Resources
for Power Macintosh and
Macintosh Programmers**
By Joe Zobkiw
Addison-Wesley, 1995. ISBN 0-201-48358-0.
500 pages (softback with CD ROM). $39.95.

Have you looked at how big the complete set of *Inside Macintosh* books is nowadays? My incomplete set is over two feet wide and growing. (I prefer paper to on-line help.) Does the very voluminosity of this documentation, as Frederick Brooks of *Mythical Man-Month* fame would say, represent some new kind of incomprehensibility? I'm not sure. While I find *Inside Macintosh* indispensable, I also find it somewhat unwieldy at times. Fortunately, there are other, more readable sources of truth available. *A Fragment of Your Imagination* is one such source.

This book is a kind of a "Cliff's Notes" for the Code Fragment Manager. It presents you with discussion and source code for any type of code fragment you might want to write: LDEFs, CDEFs, WDEFs, INITs, XMCDs, cdevs, thngs, plug-ins, etc. As do most programming books, it comes with a CD-ROM bearing all the source code in the book plus a version of CodeWarrior Lite.

There's little doubt that anyone who wants to write a 68K, PPC or fat code fragment would find it useful to have a piece of working sample code to start with. That's the primary purpose of this book. It gives you examples of each type of code fragment (LDEF, CDEF, etc.) for each environment (68K, PPC, fat). When you think about it, there's not much that can be said on the subject of stand-alone code fragments. Once you've written one or two you pretty much get it and you're on your way. But it's nice to have some hand-holding for that first one, and that's what this book does.

I like the fact that this book focuses on useful examples. The entire second half of the book is source code listings. The first few chapters cover the basics of the Mixed Mode Manager, Universal Procedure Pointers and Shared Libraries. After that, the author gets right into it by showing you how to write a fat INIT called MenuScript that allows you to replace any menu item in any program with an AppleScript. Each of the next eight chapters covers a different type of code resource, explaining the special cases that come up for each kind of code fragment, along with some example source on how to deal with it.

The examples in the book are reasonable. They are one or two steps above minimum working examples, which is what you want in a code example. If you're trying to learn how to patch a trap, for instance, you don't need 1000 lines of unrelated code. You just want the minimum to get the job done, plus a little visual feedback code to make sure it's being called. The author starts most sections by giving you the bare minimum of code necessary to write a particular type of code fragment. He explains how this bit of glue code makes the code fragment work. Then he adds a bit of unrelated code that does something visual so you'll know the code is being called and working. It's a good way to learn what the minimum code for a particular fragment type is.

While I liked the straightforward examples, completeness and screen shots, there are some things I did not like. There are too many cute expressions and attempts at humor. For example: "The termination routine couldn't be much simpler without breaking a law or two." I guess the author is trying to make the book friendly or funny, but there are so many of these lines that they come off as a bit childish.

After reading the book I couldn't help but think that it was physically much larger than it needed to be. The outer margins

TCP/IP Scripting Addition

The Internet Scripting Solution

The **TCP/IP Scripting Addition** allows you to quickly develop Internet client/server applications using AppleScript®. If you want to script with MacTCP ™ and Open Transport™, here's your solution!

- Supports Script Editor, FaceSpan ™, and HyperCard™
- Build net–wise WebSTAR™ CGI scripts and NetScape™ CCI scripts
- Sample scripts include FTP, Gopher, Telnet, Post Office, E–Mail and more
- Featured on the Apple® Internet Server

ONLY $49 SRP

Order now through the MacTech Mail Order Store at 805–494–9797 or other mail order stores

Mango Tree Software, Inc.
Box 1057 • Brookline, Massachusetts 02146
617–327–8663 • http://www.mangotree.com

All trademarks are properties of their respective holders.
Contact Mango Tree Software for site licensing and redistribution information.
Copyright © 1996 Mango Tree Software, Inc.

are over two inches wide, but that space is never used for anything. And the leading (vertical space between lines) in the source code examples is too large. I'd rather see more lines of code on the same page so I don't have to turn so many pages when studying it. By the same token, the commenting style he uses in the second half of the book really spaces the code out vertically. I'd rather have it more compact. Overall, I couldn't help but feel that the author was told to write a 500-page book and was only able to produce 300 or so pages.

Despite my nit-picking, I have to say this book does the job it set out to do – inform Macintosh programmers (who have a bit of Mac programming experience already) on how to write many kinds of code fragments. It's a nice distilled version of the relevant tech notes, DTS examples, and *Inside Macintosh.* I would definitely recommend it for your first or second code fragment. But if you've already done a few of those you probably won't get much out of this book.

Good Things Come in SMALLER Packages

Why have more than 700 developers selected Smaller Installer to deliver their products to customers around the world?

- ✓ Simple user interface
- ✓ Customizable windows and splash screens
- ✓ No scripting required
- ✓ Compact Pro® compression technology minimizes number of diskettes required
- ✓ Installer uses less than 40K of disk space
- ✓ Locates and installs files in system-related folders (Extensions, Fonts, etc.)
- ✓ Converts fat binaries to machine-specific versions during install
- ✓ Uninstall capability
- ✓ Password protection
- ✓ Extendable operation using your custom code resources
- ✓ Foreign language support – French, German, Japanese and Spanish

Smaller Installer is licensed with a one-time fee. No annual fees. Prices vary from $175 to $1,000 depending on quantity shipped.

Call or visit our web site to obtain a free developer kit which includes everything necessary for creating your own installers.

Smaller Installer™

Cyclos
P.O. Box 31417
San Francisco, CA 94131
415/821-1448
415/821-1168 (fax)
sales@cyclos.com
http://www.cyclos.com

JAVATECH™

By Kevin M. Savetz

Getting the Jump on JavaScript

Start learning to add life to your Web pages

Static Web pages are a dying breed. Although not obsolete yet, it may be one day soon that a Web page that doesn't do anything just won't get a second glance. Once upon a time, all Web pages were static. CGI programs started to change that by allowing powerful searches of Web pages, sites that modified themselves based on the time of day or the phase of the moon, and even interactive games. Java, a powerful programming language, is bringing a whole new level of interactivity to the Web. Somewhere in the middle is JavaScript, a new language that's simple enough for beginning programmers to learn, but powerful enough to interact with Web pages and Java applets in ways previously not possible with any language.

JavaScript is a fledgling language. As I write this, JavaScript isn't even complete. The technical specifications of the language have been agreed upon, but no Web browsers completely support JavaScript. (JavaScript 3.0, which should incorporate the functions that are currently lacking, is due out in June, give or take a month; no firm date had been set at press time.) But that doesn't mean that no one is using it; bleeding-edge Web hackers have been experimenting with the crippled JavaScript for months, and many are impressed with what they see.

WHAT IS JAVASCRIPT?

Java and JavaScript are distinct languages, with different purposes and features. JavaScript was designed to provide an easy way for Web authors to create interactive Web pages. Java is a programming language which is used to create stand-alone applications, called applets. Unlike Java, which is meant for experienced programmers with an understanding of C++, JavaScript is a simpler "scripting" language (like dBASE and AppleScript) aimed at those with less programming experience.

Both are powerful languages, but they each offer distinct functionalities. According to Andy Augustine, vice president of technology for Frequency Graphics, a Web development and hosting service provider, "JavaScript is aimed at the people who don't have C++ experience, people who don't have much experience with CGI, but want to create some sort of interactive Web page."

"Contrary to popular misconceptions, JavaScript was not meant to be a scaled-down version of Java, nor was it intended to be a replacement for CGI (server-side) scripts. Instead, JavaScript functions as an outstanding way to enhance both," he said.

JavaScript can be used to manage user input as well as to show text, play sounds, display images, or communicate with a plug-in in response to "events" such as a mouse-click or exiting or entering a Web page.

JavaScript's syntax is loosely based on the Java language, but those who have used both languages say the similarity is very loose indeed. In fact, the closest resemblance the two languages have is their names. JavaScript was originally called Mocha while being developed at Netscape. It was later dubbed LiveScript, and when Netscape Communications partnered with Sun to develop Java, LiveScript was renamed JavaScript.

Kevin Savetz is author of *MBONE: Multicasting Tomorrow's Internet*, published by IDG books and waiting patiently on a bookshelf near you. You can visit Kevin's Web page at `http://www.northcoast.com/savetz/` or reach him at savetz@northcoast.com.

JAVA VS. JAVASCRIPT

JavaScript programs are interpreted and run entirely on the client side. This means fewer hits and less processing time on the server than with Java (where applets are compiled on the server before being executed on the client) or CGI (which requires the server to do the work and rack up hits).

There are other differences between the two languages. In Java, applets are files distinct from HTML pages; with JavaScript, the code is integrated into the HTML using special tags. (This can cause problems with some Web browsers that do not properly handle JavaScript code.) Like Java, JavaScript is a cross-platform language that can work with any compatible browser.

Here's an example HTML file with an embedded JavaScript. Between the `<BODY>` tags, the Web browser displays a form with a space to enter a number. When the user enters a number in the field, the browser starts the script, located between the `<SCRIPT>` tags. The script itself makes sure the user really entered a number, and displays a dialog box if the user didn't keep the number within pre-defined limits.

```
<HEAD>
<SCRIPT LANGUAGE="JavaScript">
<!--- these comments should hide the script from old browsers
function checkNum(str, min, max) {
  if (str == "") {
    alert("Enter a number in the field, please.")
    return false
  }
  for (var i = 0; i < str.length; i++) {
    var ch = str.substring(i, i + 1)
    if (ch < "0" || ch > "9") {
      alert("Enter a number, please.")
      return false
    }
  }

  var val = parseInt(str, 10)
  if ((val < min) || (val > max)) {
    alert("Try a number from 1 to 10.")
    return false
  }
  return true
}

function thanks() {
  alert("Thank you.")
}
// end hiding from old browsers -->
</SCRIPT>
</HEAD>

<BODY>
<FORM NAME="ex5">
Please enter a number from 1 to 10:
<INPUT NAME="num"
  onChange="if (!checkNum(this.value, 1, 10))
    {this.focus();this.select();} else {thanks()}">
</FORM>
</BODY>
```

All this only takes one hit on the server – to download the HTML code and the embedded script. Doing this with a CGI script would require multiple hits on the server, once for the HTML file, and again to verify the input every time the user types into the form. JavaScript allows the client to perform more significant calculations without weighing down the server.

Other differences are more technical. JavaScript is object-based: its code uses built-in, extensible objects, but unlike Java, there are no classes or object inheritance. In JavaScript, object references are checked at run-time; in Java, object references must exist upon compilation. In JavaScript, variable data types are not declared, unlike in Java, where they must be declared before use.

Which language is better? Until a JavaScript interpreter is complete, there is no fair comparison. Although it was originally intended as a tool for fledgling programmers, even experienced programmers who know Java will find JavaScript useful. The languages can work together to complement each other.

Currently, the only Web browser with any support for JavaScript is Netscape 2.0, and it lacks much of the functionality promised for the final release. More than 30 other companies, including Microsoft, have agreed to include support for JavaScript in their Web browsers. With the current version of JavaScript, you can send information to the browser (such as a dialog box) and get the status of objects (such as mouse movements), but some of the most exciting functions, such as the abilities to read input streams from documents and to interact with Java applets, are still unimplemented.

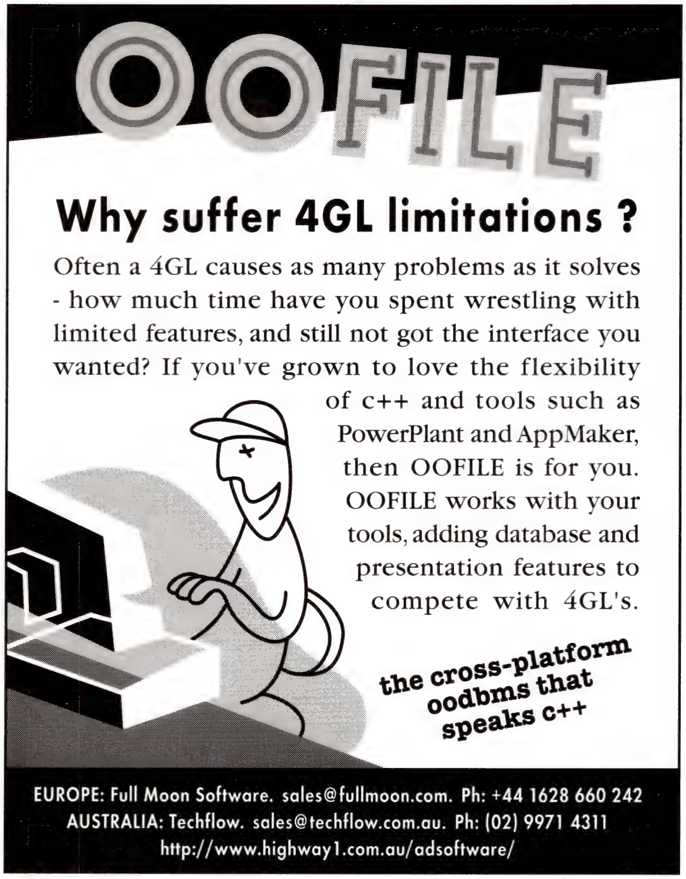
When JavaScript includes all of the tools the specs say will be included, including something called the "applet object" (which will allow scripts to interact with Java applets), the two languages will be a mighty team. "As soon as the applet object is released, you will be able to control Java applets with JavaScript, and that will make them work perfectly together," Augustine says. This means that you will be able to control a Java applet using a Web-based form, with a JavaScript doing the behind-the-scenes translation between the Web page and the applet. The user will be able to use a radio button to select how a Java applet will run – for instance, changing the color and speed of a rotating logo. JavaScript will pass information from the form to the Java applet. JavaScript's applet object will also allow scripts to send information to and get information from other scripts and CGI programs.

Another tool that will be available in the final release of JavaScript is the "history object", which will allow the script to see what Web pages the user has recently visited (although to protect the user's privacy, the script will not be able to share that information with the server). This would allow a page to offer an intelligent "back" button, letting the user move quickly back to any previously visited Web page.

As with Java, security in JavaScript is likely to be an ongoing battle. Several security problems were quickly discovered in JavaScript 2, including a bug that let JavaScript send a blank electronic mail message to someone without the sender's knowledge. Another bug allowed JavaScript to get a listing of files on the user's hard drive (although the script could not read the contents of any files).

NOW OR LATER?

Should you bother learning JavaScript now, or is it currently just a toy for experimenters? Should you wait until the final


OOFILE
the cross-platform oodbms that speaks c++

Why suffer 4GL limitations ?

Often a 4GL causes as many problems as it solves - how much time have you spent wrestling with limited features, and still not got the interface you wanted? If you've grown to love the flexibility of c++ and tools such as PowerPlant and AppMaker, then OOFILE is for you. OOFILE works with your tools, adding database and presentation features to compete with 4GL's.

EUROPE: Full Moon Software. sales@fullmoon.com. Ph: +44 1628 660 242
AUSTRALIA: Techflow. sales@techflow.com.au. Ph: (02) 9971 4311
http://www.highway1.com.au/adsoftware/

version of the JavaScript interpreter ships before trying to learn the ins and outs of the language? Augustine thinks that now is the time to get a jump start on JavaScripting. "Learn JavaScript right now if you want to capture the market while it is viable; you'll be ahead of the field. If you wait until it is fully functional before you learn it, I think you'll be behind the game," he says. Online tutorials can get you started with the basics of the language, and Internet mailing lists and newsgroups devoted to the topic are already home to thriving communities of developers and experimenters.

Will the serious Web page developer need to understand both Java and JavaScript? Because JavaScript can interact directly with the user anywhere on a Web page (rather than being limited to a single window, as with Java), most Java developers will probably want to learn both. But JavaScript should also be viable as a stand-alone tool for less experienced programmers who want to add interactivity to their Web pages.

EXAMPLE SCRIPTS

There are already dozens of example JavaScripts on the Web. Many, such as the 1040EZ tax form and body mass calculator, simply compute values input from an HTML form. Others are more inspired, such as the currency exchange calculator, which lets you select two countries from a list and then computes the exchange rate. I am particularly fond of the Zen koan generator, even though it occasionally causes Netscape 2.0 to crash.

http://www.homepages.com/fun/1040EZ.html
http://www.iohk.com/UserPages/acheng/javascript.html
http://www.superprism.net/doc/EXPRESS/util/currency.html
http://www.wrldpwr.com/javascriptzen.html

The color picker shows how JavaScript can work with frames and color: pass the mouse over the name of a color in one frame and the background of another frame will become that color. Other examples are downright inspirational, such as the JavaScript VRML clock, which uses JavaScript to display a three-dimensional clock if you have a VRML browser.

http://www.lsi.usp.br/doc/color/picker.html
http://www.phoenix.net/~jalonso/applets/vrml

Many more example applications are available at http://www.gamelan.com/.

Remember how interesting (and sometimes annoyingly funky) Web pages became when authors started using and abusing sundry new tools like backgrounds and frames? Expect to see another round of enlightening and obscure Web sites as developers and hackers begin to understand and use JavaScript.

FOR MORE INFORMATION

A growing number of information sources about JavaScript are online. Andy Augustine's JavaScript FAQ is at:

http://www.freqgrafx.com/411/jsfaq.html

There is also a JavaScript mailing list, a high-volume, sometimes technical list for discussion about using the language. To subscribe, send a message to majordomo@obscure.org with a message body of "subscribe javascript". A digest version of the list is also available. To subscribe, use a message body of "subscribe javascript-digest". Further discussion takes place on the newsgroup comp.lang.javascript.

Netscape's official JavaScript documentation is required reading if you plan to learn your way around JavaScript. The files are also downloadable for reading offline.

http://home.netscape.com/eng/mozilla/2.0/handbook/javascript/index.html
http://home.netscape.com/eng/mozilla/2.0/handbook/javascript/jsdoc.zip

Augustine maintains a list of online tutorials, example applications and other information at JavaScript 411. Finally, the ubiquitous Yahoo naturally has an index of JavaScript-related goodies.

http://www.freqgrafx.com/411/
http://www.yahoo.com/Computers_and_Internet/Languages/JavaScript/

UNIFORM RESOURCE LOCATORS

By Jim Straus, URLs@mactech.com

Welcome to the latest update on Macintosh/Internet information. This month we have several Internet magazines or e-zines as well as information on Apple's Game Sprockets APIs. As always, there is Internet stuff and programmer resources at your finger tips. Or spare your fingers and find the full list online at:
http://www.mactech.com/URLs.html

Latest Updates

Internet Related

Mac Internet Server Software	http://www.cen.uiuc.edu/~wronkiew/macos_servers/
MacJava Development	http://www1.omi.tulane.edu/adowns/decaf
MacJava Discussion Group	http://www.mactivity.com/java/java_discussion.html

New Technologies

Game Sprockets	http://dev.info.apple.com/evangelism/games/games.html

Other Programmer Resources

Conflict Compendium	http://www.islandnet.com/~quill/c3data.html
Emerald Ventures	http://www.dataphone.se/~punt/emve.html
Mac Hacking and Cracking	http://www.netaxs.com/~hager/machack.html
OpenDoc eZine	http://www.partmerchant.com/odz/
PowerPlant Beginners	http://www.interlog.com/~breakpt/html/powerplant.html

Vendors, Products and Miscellaneous

Absolute Mac	http://www.ie2.u-psud.fr/~peirano/mac_us.html
AppMaker	http://members.aol.com/bowersdev/
Casady & Greene	http://205.179.133.2/C&G/Welcome.html
Mac-o-mania	http://remus.rutgers.edu/~euge/mac.html
Mac Related Sites	http://www.net-army.com/mac/
MacSense Magazine	http://www.macsense.com/MacSense/
MacUser Magazine	http://www.zdnet.com/macuser/
Read Me 1st	http://www.idir.net/~jgamble/

Internet

If you want to serve information on the Internet, and are not sure what you will need, you should check out Mac Internet Server Software. It lists Web servers, ftp servers, gopher servers, mail servers, telnet servers, finger servers, domain name servers – anything you could ask for. Each server is rated for compatibility with MacTCP and Open Transport. Also included are some links useful for people setting up servers. An excellent resource.
Mac Internet Server Software
http://www.cen.uiuc.edu/~wronkiew/macos_servers/

It seems that we cannot go a month without more Java sites. I am not going to say much about them, just mention them.
MacJava Development **http://www1.omi.tulane.edu/adowns/decaf**

MacJava Discussion Group
http://www.mactivity.com/java/java_discussion.html

Macintosh

First off, if you have not seen it yet, you need to look in on the Apple's Game Sprockets site. This is not the Sprocket™ application framework by *MacTech Magazine*, but the Apple's Game Sprockets APIs. Included are Network Sprockets, QuickDraw 3D Sprockets, Speech Sprockets, Draw Sprockets, Sound Sprockets and Input Sprockets. "Sure," you say, "these are just *Inside Macintosh*." Actually, they are SDKs with sample applications, docs, and project frameworks. If you are interested in games on the Mac (or on the Pippin), this is the place for you.
Sprockets (Game APIs)
http://dev.info.apple.com/evangelism/games/games.html

If you are looking for more OpenDoc information, read OpenDoc eZine (ODZ). This is a nice-looking Web-based publication that is devoted to OpenDoc. Be warned, it covers not only OpenDoc for the Macintosh, but the other OpenDoc platforms as well.
OpenDoc eZine **http://www.partmerchant.com/odz/**

For a more mainstream Macintosh e-zine, check out MacSense. It looks to be professionally assembled, and has up-to-the-minute information. Besides pointers to the latest news, there are reviews, columns, editorials, and even letters to the editor. This is a great source for learning about what is going on in the Mac community.
MacSense Magazine **http://www.macsense.com/MacSense/**

Neat non-Macintosh site of the Month

Whether you are a beginner or an expert Internet user, sometimes you will run into trouble and need help. Help can be found at the Internet Help Desk. If you are getting error messages back when trying to read mail, or problems fetching a file, this is the spot. Even if you are not confused by something, you should check in some time. You might learn some new way to do something, or learn about some aspect of the Internet unknown to you. It is not Macintosh-specific, but it has lots of help. A great reference!
The Internet Help Desk **http://w3.one.net/~alward/**

Well, that is it for this month. As always, if you find something interesting, or have updates, send them to URLs@MacTech.com.

Thanks this month to Spec Bowers, Jonathan Hirshon, Lukas Hjalmar, Darren King, Peirano Philippe, David L. Slipp, Matt Wronkiewicz, and many others for their contributions for their suggestions and pointers to new and old sites.

PROGRAMMING TOOLS

By Rich Parker

Tools Plus 3.0

Paradise for "control" freaks

EVERYONE WANTS TO PROGRAM THE MAC

Almost all Mac owners, at one point or another, harbor a desire to program their own machine. Some may be well served by products like HyperCard, but others just can't wait to learn C, C++, or Pascal, so that they can write the next "killer" application.

If they start programming by buying books that illustrate programs that use the standard ANSI libraries, they might make quite a bit of progress in learning the language and writing routines that use a "console" interface. At some point, though, the realization sinks in that these aren't "real" Mac applications, and then the trouble begins. The age-old question that I see over and over again on the online services is: "How do I get started writing 'real' Mac programs?"

Most people are directed to read one of Dave Mark's or Dan Sydow's fine books on the subject of Mac programming for their favorite language, but these books provide only the basic information needed to write "real" Mac programs. At some point, the (now quite distressed) user is directed to the almighty gospel of *Inside Macintosh* to further his or her knowledge. This is the point at which most budding programmers wimp out, and I don't really blame them. I think you will agree that trying to learn to program the Mac by reading *Inside Macintosh* is like learning to play the piano by studying the details of music composition.

SO WHAT'S THE ALTERNATIVE?

I think it's well known in the developer community that I favor the use of application frameworks. Developing Mac applications using PowerPlant, TCL, or MacApp is something that comes easily to me. But I also recognize the negative aspects of these frameworks, and readily admit that they are not a solution for all people or all applications.

In the first place, most application frameworks demand much more memory than would be required if one had the time to read, understand, and apply all of the appropriate built-in Mac toolbox routines to the problem. But the difficulty of understanding the Mac toolbox is equaled only by the steep learning curve of most of the current application frameworks. So we are left with an almost insoluble problem – whether to spend endless hours reading *Inside Macintosh* or to spend equally endless hours studying the structure and functionality of an application framework.

TOOLS PLUS TO THE RESCUE!

Steve Makohin, Steven Waters, and the folks at Water's Edge Software realized that programming the Mac wasn't easy. After all, they ran into these problems themselves when they first started. So they set out to create a set of high-level functions that would enable others to program the Mac much more easily, without having to worry about handling some of the more difficult aspects of Mac programming. The result was a set of binary libraries that could be included in projects for your

Rich Parker is a book author and software developer in Modesto, CA. He spends his time writing programming books for both the Macintosh and Windows platforms. He is currently involved in a new book project and has little free time. He is also a lover of all computer languages, especially object-oriented dynamic languages. His latest book for the Mac, *Mastering the THINK Class Library*, is available at technical bookstores and through Developer Depot. If you would like to write to Rich, asking a question or correcting a mistake, please feel free to do so. Rich can be reached on CompuServe at 76556,1132 or on the Internet at rparker@s2.sonnet.com.

favorite development environment, whose routines could be called to implement and manage exceedingly sophisticated Mac user interfaces. (After all, it's the user interface that is the most complex aspect of any Mac application development.)

The first Tools Plus product was developed as a result of an R&D project in 1989. By 1992, after considerable prodding from various contractors and project managers, Makohin and Waters decided to polish their product to commercial readiness. A year later, Water's Edge Software was formalized to become the delivery vehicle for Tools Plus, and to support its users.

The product has been continually enhanced, and version 3.0 of the Tools Plus product is even more capable than any previous version. It comes with a wealth of features that promise to make almost any Mac programming project a pleasure, rather than a pain.

WHAT'S IT ALL ABOUT?

Tools Plus version 3.0 consists of a set of GUI Building/Event-Managing libraries for Macintosh programmers using any of the following compilers:

- Metrowerks CodeWarrior C/C++ and CodeWarrior Pascal for either 680x0-based or Power Macintosh systems. All versions of CodeWarrior from 6.0 through 8.0 are supported. The Power Mac libraries are native for both C/C++ and Pascal.
- THINK Pascal version 4.0.2 or later for 68K development only.
- THINK C version 5.0.4 or later for 68K development only.
- Symantec C/C++ version 6.0.1 or later for 68K development only. This includes the newly released Symantec version 8.1, which supports 68K development with the Symantec Project Manager; however, in contrast to the CodeWarrior situation, where the PPC-native Tools Plus libraries have been available since mid-1995, native Power Macintosh libraries are not yet available in version 3.0 of Tools Plus for the SPM. [*Portions of the Tools Plus source are in Pascal, and until recently there wasn't SPM PPC-native Pascal support. Water's Edge feels that the LS Pascal translator needs some maturing before they can release the code it generates. – man*]

To use the Tools Plus product, you include the libraries and the `ToolsPlus.c` file in your Symantec or Metrowerks project (the `ToolsPlus.h` file is also needed). In the case of Pascal-based projects, the `ToolsPlus.p` file is added to the project along with the ToolsPlus libraries. Starter project files and all other requisite files (e.g., `ToolsPlus.c`, `ToolsPlus.h`, `ToolsPlus.p`) are provided in the product that you purchase. The products offered by Water's Edge are as follows:

- Libraries for Symantec compilers (68K only), including THINK C and SPM v8.1 ($149).
- Libraries for the THINK Pascal compiler for 68K development ($149).
- Libraries for both THINK C/SPM and THINK Pascal for 68K development ($199).
- Libraries for Metrowerks CodeWarrior Bronze (68K only), including C, C++, and Pascal ($199).
- Libraries for Metrowerks CodeWarrior Gold (68K and PPC), including C, C++, and Pascal ($249).
- Tools Plus v3.0 Academic for any of the foregoing environments ($99 each).
- Super CDEFs, a collection of special "Control Definition" resources that you can include in your applications for any Macintosh ($89 when purchased alone, $39 for those with an academic discount, or free with the purchase of any non-academic Tools Plus developer kit).

WHAT FEATURES DO THE LIBRARIES PROVIDE?

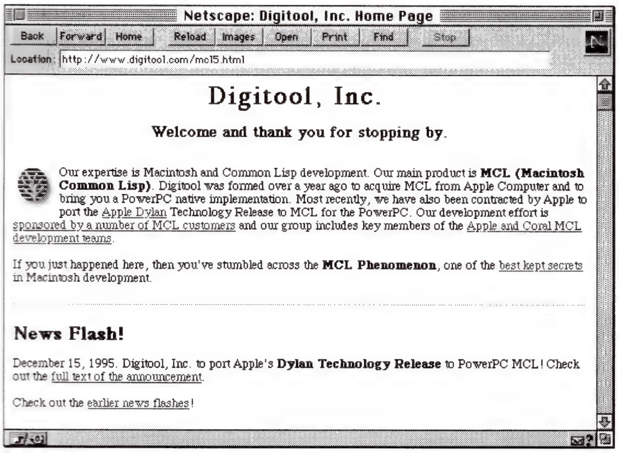
The Tools Plus libraries provide a comprehensive set of user interface elements and event-management features. Every control is created with a single statement in your application. The user interface elements include windows of various types, including all standard window types, floating palettes, movable modal dialogs, buttons, editing fields, raised or sunken 3D panels, and many other features. The best way to show you some of the things you're likely to use is to display some screen shots of the Demo program that ships with every copy of the Tools Plus product, and is also available on all of the major

Macintosh® Common Lisp **Digitool**

An Object-oriented Dynamic Language

Now PowerPC Native!

For the full story visit our Web site:
http://www.digitool.com/mcl5.html



Netscape: Digitool, Inc. Home Page
Back
Forward
Home
Reload
Images
Open
Print
Find
Stop
Location: http://www.digitool.com/mcl5.html

Digitool, Inc.

Welcome and thank you for stopping by.

Our expertise is Macintosh and Common Lisp development. Our main product is **MCL (Macintosh Common Lisp)**. Digitool was formed over a year ago to acquire MCL from Apple Computer and to bring you a PowerPC native implementation. Most recently, we have also been contracted by Apple to port the Apple Dylan Technology Release to MCL for the PowerPC. Our development effort is sponsored by a number of MCL customers and our group includes key members of the Apple and Coral MCL development teams.

If you just happened here, then you've stumbled across the **MCL Phenomenon**, one of the best kept secrets in Macintosh development.

News Flash!

December 15, 1995. Digitool, Inc. to port Apple's **Dylan Technology Release** to PowerPC MCL! Check out the full text of the announcement.

Check out the earlier news flashes!

Digitool, Inc. • One Main Street • Cambridge, MA 02142
tel: (617) 441-5000 • fax: (617) 576-7680 • email: info@digitool.com

Help Wanted

Join the best virtual engineering team in the industry! The Now Utilities™ Team needs engineers and utilities for:

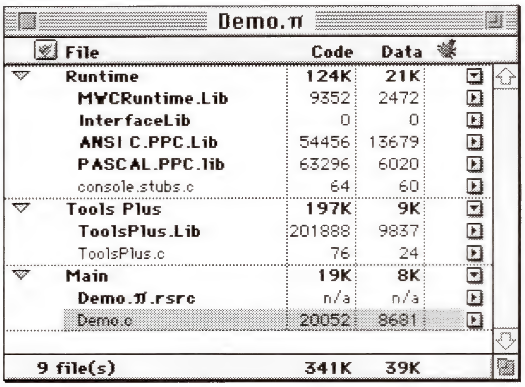
- **Now Utilities 7.0**
  We need new utilities and engineers to enhance existing utilities
- **Now Utilties for Copland**
  We need new utilities for Copland
- **Now Utilities Plug Ins™**
  We need new Plug Ins for Now Utilities 6.0, 7.0 and Now Utilities for Copland

Join our virtual engineering team and we'll give you the exposure, recognition and royalties only Now Utilities distribution can provide.

Now Software
http://www.nowsoft.com

For more information on becoming part of the Now Utilities Team send mail to: utilities@nowsoft.com.

online services (along with its source code). I built the demo application inside CodeWarrior version 8.0 for the PowerMac. The project file for that build is shown in Figure 1. You can see from the figure that the Demo program's project file contains several of the standard Metrowerks libraries, as well as the `Tools Plus.Lib` library. The Demo program is 228K in size and runs in a partition of 384K. It includes examples of many of the features that are available in the Tools Plus product.

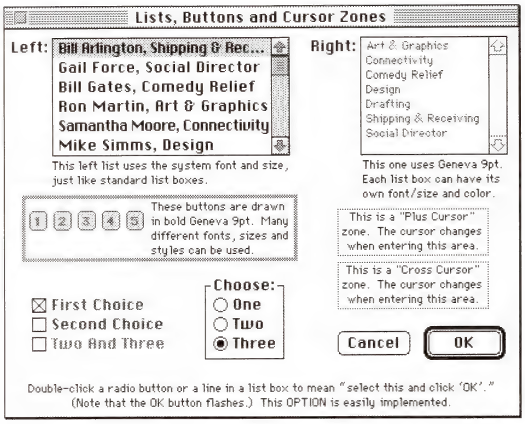


Demo.π

File	Code	Data
▽ **Runtime**	**124K**	**21K**
MWCRuntime.Lib	9352	2472
InterfaceLib	0	0
ANSI C.PPC.Lib	54456	13679
PASCAL.PPC.lib	63296	6020
console.stubs.c	64	60
▽ **Tools Plus**	**197K**	**9K**
ToolsPlus.Lib	201888	9837
ToolsPlus.c	76	24
▽ **Main**	**19K**	**8K**
Demo.π.rsrc	n/a	n/a
Demo.c	20052	8681
9 file(s)	**341K**	**39K**

Figure 1. CodeWarrior project file for Demo program

The window in Figure 2 describes itself. When you move the cursor into the "Plus" cursor zone, the cursor changes into a plus-sign. When you move the cursor into the "Cross" cursor zone, the cursor changes into a cross. The lists demonstrate the use of different fonts with text in any selected color, and the buttons demonstrate the ability to create both standard and non-standard buttons. The checkboxes in the lower left corner demonstrate the ability to disable the "Two And Three" checkbox while the "Second Choice" checkbox is unchecked.



Lists, Buttons and Cursor Zones

Left:
Bill Arlington, Shipping & Rec...
Gail Force, Social Director
Bill Gates, Comedy Relief
Ron Martin, Art & Graphics
Samantha Moore, Connectivity
Mike Simms, Design

This left list uses the system font and size, just like standard list boxes.

Right:
Art & Graphics
Connectivity
Comedy Relief
Design
Drafting
Shipping & Receiving
Social Director

This one uses Geneva 9pt. Each list box can have its own font/size and color.

These buttons are drawn in bold Geneva 9pt. Many different fonts, sizes and styles can be used.

This is a "Plus Cursor" zone. The cursor changes when entering this area.

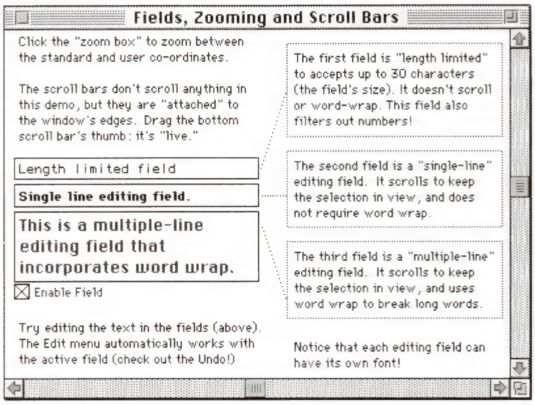
This is a "Cross Cursor" zone. The cursor changes when entering this area.

☒ First Choice
☐ Second Choice
☐ Two And Three

Choose:
○ One
○ Two
● Three

Double-click a radio button or a line in a list box to mean "select this and click 'OK'." (Note that the OK button flashes.) This OPTION is easily implemented.

Figure 2. One of the Demo program's feature demonstrations



Fields, Zooming and Scroll Bars

Click the "zoom box" to zoom between the standard and user co-ordinates.

The scroll bars don't scroll anything in this demo, but they are "attached" to the window's edges. Drag the bottom scroll bar's thumb: it's "live."

Length limited field

Single line editing field.

This is a multiple-line editing field that incorporates word wrap.

☒ Enable Field

Try editing the text in the fields (above). The Edit menu automatically works with the active field (check out the Undo!)

The first field is "length limited" to accepts up to 30 characters (the field's size). It doesn't scroll or word-wrap. This field also filters out numbers!

The second field is a "single-line" editing field. It scrolls to keep the selection in view, and does not require word wrap.

The third field is a "multiple-line" editing field. It scrolls to keep the selection in view, and uses word wrap to break long words.

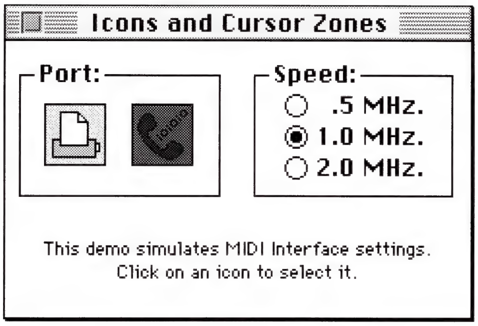
Notice that each editing field can have its own font!

Figure 3. More features from the Demo program

In Figure 3, various types of text field are shown. It also demonstrates the use of "live" scroll bars: instead of displaying a "ghost" outline of the thumb when it is dragged, the thumb of the horizontal scroll bar is redrawn in real time as you drag it. (The thumb of the vertical scroll bar displays the normal appearance of an outline when the thumb is dragged.)

Figure 4 simulates a setup window for a MIDI interface, with the two icon buttons selecting a port and the three radio buttons selecting the interface's speed. (Despite the window's name, I didn't see any "Cursor Zones" in this example; the cursor remained in the shape of an arrow within the window.)

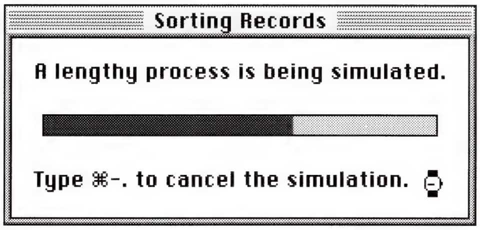
In Figure 5 the watch cursor is animated while the progress indicator ("thermometer") is advancing.



○ .5 MHz.
● 1.0 MHz.
○ 2.0 MHz.

This demo simulates MIDI Interface settings.
Click on an icon to select it.

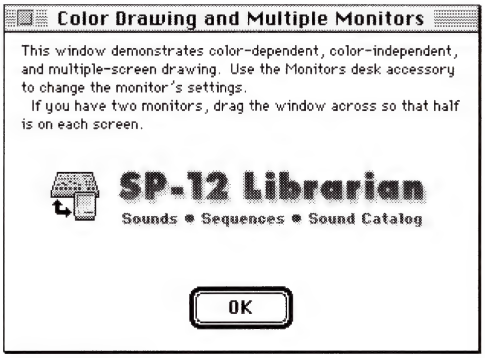
Figure 4. Icon button example



A lengthy process is being simulated.

Type ⌘-. to cancel the simulation.

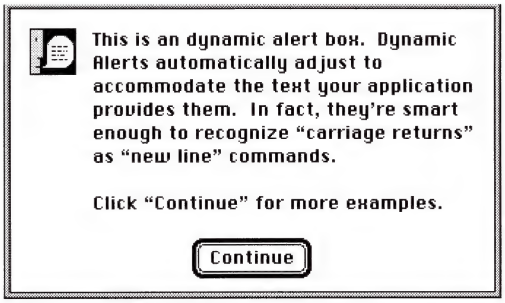
Figure 5. Pretending to sort


Color Drawing and Multiple Monitors
SP-12 Librarian
Sounds • Sequences • Sound Catalog
OK

This window demonstrates color-dependent, color-independent, and multiple-screen drawing. Use the Monitors desk accessory to change the monitor's settings.

If you have two monitors, drag the window across so that half is on each screen.

Figure 6. Color-independent drawing



This is an dynamic alert box. Dynamic Alerts automatically adjust to accommodate the text your application provides them. In fact, they're smart enough to recognize "carriage returns" as "new line" commands.

Click "Continue" for more examples.

Figure 7. Self-sizing alert

Figure 6 illustrates how the Tools Plus library compensates for the color abilities (or lack thereof) of your Macintosh. When the figure is drawn on a monochrome monitor, the picture and accompanying text are drawn so that they are still entirely readable. I didn't have two monitors to demonstrate to myself

DragInstall® 2.0

Here's proof

that the more things change

- ▲ New "Quick Build" option lets you build a complete installer in *one easy step*.
- ▲ New features such as locating files and folders, applying patches, and replacing outdated files allow you to build more intelligent installers.
- ▲ Improved support for installing non-archived files simplifies the creation of CD-ROM and network installers.
- ▲ PowerPC-native compression and decompression cuts installation time *in half.*
- ▲ AppleEvent and scripting support allows installations to be automated.

the more they stay the same

- ▼ Same price–$300 lets you distribute an *unlimited* number of installers for *all* of your products. No yearly fee, no royalties, no hidden costs.
- ▼ Same drag-and-drop interface familiar to all Macintosh users.
- ▼ Same reliability and robustness that developers worldwide have relied on since 1991.
- ▼ Same support policy–free technical support and low (or no) cost upgrades.

For more information or a free demo, call
1-800-890-9880
or visit our Web site at
http://www.sauers.com/draginstall

Ray Sauers Associates

Ray Sauers Associates
1187 Main Avenue, Suite 1B
Clifton, NJ 07011 USA
voice: 201-478-1970
fax: 201-478-1513
email: info@sauers.com

World's Leading FORTRAN 77 for Macintosh

68K and Power Macintosh versions

- full ANSI 77 native compiler
- faster execution speed
- graphical source-level debugging
- two complete graphics packages
- make utility, MRWE application framework
- System 7.5 compatible, MPW included
- Windows 95/NT version also available

absoft
development tools and languages

Visit us now at http://www.absoft.com

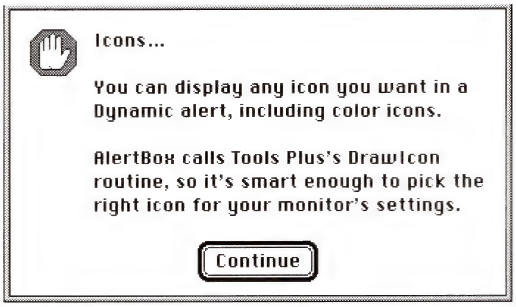
2781 Bond Street Rochester Hills MI 48309 • (810) 853-0050 • Fax (810) 853-0108 • sales@absoft.com

that one-half of the window would be drawn in monochrome and the other in color, but I did change the Monitors control panel settings to verify that the color text and the picture are perfectly readable when using the "Black & White" setting.

The alert shown in Figure 7 illustrates that you can put whatever text you want displayed, including carriage-return characters, and the Tools Plus libraries will create a dialog that's exactly the right width and height to hold that text. (Carriage-return characters are specified as `'\r'` in the text string.)

The alert shown in Figure 8 demonstrates that dynamic alerts can contain any icon you choose, including the System's standard icons. After pressing the Continue button in this alert, you are given the opportunity to see other examples of alerts, even alerts with no associated buttons. Clicking the mouse inside the alert dismisses it.

The Tools Plus libraries include a variety of pop-up menus (Figure 9). These can contain words, icons and words, icons only, what Water's Edge calls "Pop-Down" menus that are used to specify immediate actions to perform, as well as pop-up menus of any size, using any font.

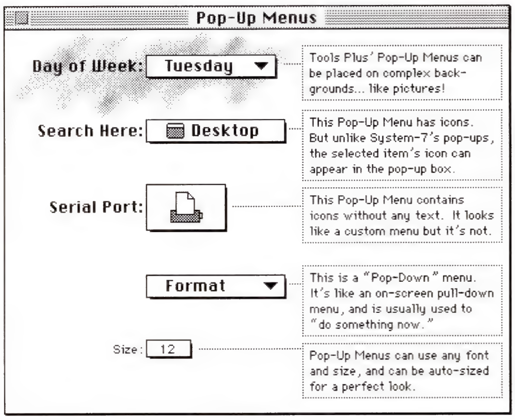


Icons...

You can display any icon you want in a Dynamic alert, including color icons.

AlertBox calls Tools Plus's DrawIcon routine, so it's smart enough to pick the right icon for your monitor's settings.

Figure 8. Self-sizing alert with icon of your choice



Tools Plus' Pop-Up Menus can be placed on complex back-grounds... like pictures!

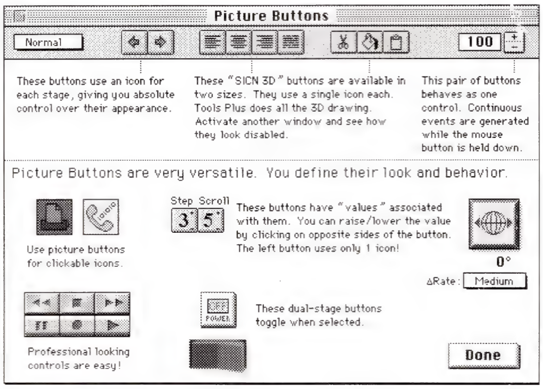
This Pop-Up Menu has icons. But unlike System-7's pop-ups, the selected item's icon can appear in the pop-up box.

This Pop-Up Menu contains icons without any text. It looks like a custom menu but it's not.

This is a "Pop-Down" menu. It's like an on-screen pull-down menu, and is usually used to "do something now."

Pop-Up Menus can use any font and size, and can be auto-sized for a perfect look.

Figure 9. Various pop-up menus



These buttons use an icon for each stage, giving you absolute control over their appearance.

These "SICN 3D" buttons are available in two sizes. They use a single icon each. Tools Plus does all the 3D drawing. Activate another window and see how they look disabled.

This pair of buttons behaves as one control. Continuous events are generated while the mouse button is held down.

Picture Buttons are very versatile. You define their look and behavior.

Use picture buttons for clickable icons.

These buttons have "values" associated with them. You can raise/lower the value by clicking on opposite sides of the button. The left button uses only 1 icon!

These dual-stage buttons toggle when selected.

Professional looking controls are easy!

Figure 10. A plethora of picture pushers

You might think that you've seen it all, but we're just getting to the good stuff. The Tools Plus libraries also contain a

Want to develop OpenDoc-compliant/Cyberdog-savvy products?

We've just made it easy.

These CD-ROMs contain the latest OpenDoc® and Cyberdog™ software—including all of the development tools and documentation needed to create OpenDoc components and Cyberdog part editors or network service components. They have everything you need to get started working with these innovative Apple® technologies.

You get...

OpenDoc

- OpenDoc DR5, which features:
  - —Complete source code for easy debugging
  - —OpenDoc Development Framework (ODF) release 1.0, the best cross-platform framework available for creating OpenDoc-compliant and Internet/Cyberdog-savvy components, parts, and containers
  - —Third-party components, including a version of ClarisWorks® that supports OpenDoc
  - —Full documentation

- Cyberdog 1.0 [the "Golden Master" (GM) release], including:
  - —Critical development materials (header files, project files, build instructions, and more)
  - —Executable samples, demos, and presentations showcasing the range of possibilities for Cyberdog development
  - —Full documentation
  - —An HTML-based tutorial that offers step-by-step instructions on how to develop a simple Cyberdog display part

CYBERDOG
Your Best Friend on the Internet

So if you want to be on the forefront of the next wave of Macintosh® software development, you've got what it takes.

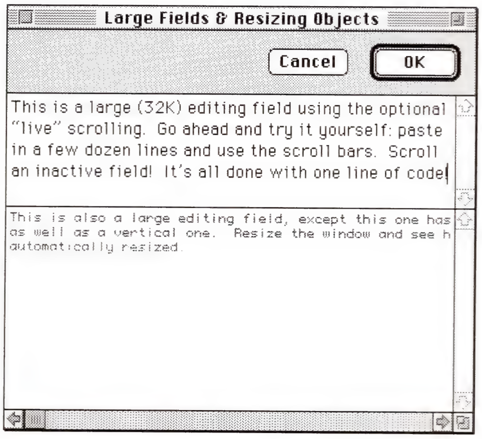
© 1996 Apple Computer, Inc. All rights reserved. Apple, the Apple logo, Macintosh, and OpenDoc are registered trademarks of Apple Computer, Inc.
Cyberdog is a tradmark of Apple Computer, Inc. ClarisWorks is a registered trademark of Claris Corporation.

large selection of picture buttons. A number of these are illustrated in Figure 10. (Sorry about the caption for the figure, but I couldn't resist!) The variety of picture buttons available in the Tools Plus libraries is just amazing. Every one of these is created by a single line of code. The 3D buttons in the lower left are examples of CD-ROM controls. The buttons at the top of the window show some of the features of text-editor and drawing programs. The middle set of buttons in the top panel illustrate various text justification styles.

The "Step" and "Scroll" buttons in the middle of the figure illustrate how the values shown on the buttons can be modified by clicking on either the top or bottom of the button. Clicking on or near the plus-symbol or minus-symbol in one of the buttons causes the numerical value in that button to increment or decrement, respectively; plus, the "Step" button changes value each time it is clicked, whereas the "Scroll" button continues to increase or decrease in value as long as the mouse button is held down.

The button at the far right, in the middle of the window, has an associated value whose minimum, initial, and maximum are specified when the button is created. If the button is clicked near its right edge, then the button's value is increased. If the button is clicked near its left edge, then the button's value is decreased. (The display of "0°" below the button is managed separately and, according to the specifications for the "Globe" button, can vary from -179° to +180°.) The button is also specified to have medium acceleration, meaning that the value changes more and more quickly the longer the button is pressed. (An acceleration rate can be specified as slow, medium, or fast.) The "Globe" button is created by a single line of code:

```
NewPictButton( GlobeButton, 410, 153, GlobeIcon,
     picbutRepeatEvents + picbutAutoValueChg +
     picbutScaleMedAccel + picbutValueWrap +
     picbutLeftRightSplit + picbutSelectAltImage +
     picbutDimUsingWhite, enabled, notSelected, -179, 0, 180);
```



This is a large (32K) editing field using the optional "live" scrolling. Go ahead and try it yourself: paste in a few dozen lines and use the scroll bars. Scroll an inactive field! It's all done with one line of code

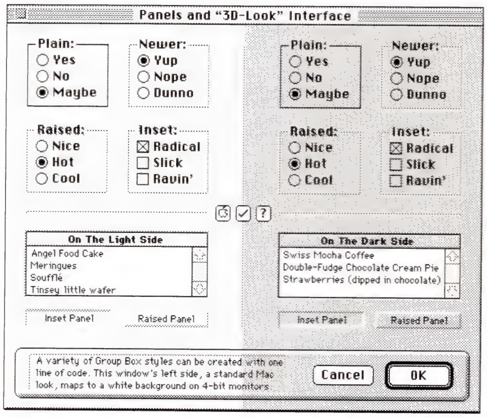
This is also a large editing field, except this one has as well as a vertical one. Resize the window and see h automatically resized.

Figure 11. Large text fields

Most of the style settings here are specified as symbolic names. The button's ID (`GlobeButton`) is a unique value that identifies the button whenever it is referenced. The Demo program defines the `GlobeButton` symbol as the value 16. The icon for the button can be anything you want and is indicated by the `GlobeIcon` symbol, which is defined to refer to a `'cicn'` resource of 460. The 410 and 153 values refer to the left and top positions of the button, and the numeric values in the last three arguments refer to the minimum, initial, and maximum values that the button can contain.

Figure 10 also shows an ON/OFF button whose caption changes whether it is on or off and also a toggle-switch which changes its appearance depending upon whether it is clicked to the left or right. It's safe to say that the figure illustrates only a few of the multitude of picture button possibilities.

Some applications require text fields that exceed the standard 255 character-width provided by Str255 variables. The Tools Plus libraries provide text fields that are up to 32K characters in width. These are illustrated by Figure 11. You can display panes that contain quite large text fields. The lower pane in the figure illustrates a field that has both vertical and horizontal scroll bars. Each of the text panes has its own font, size, style, color, and justification. The horizontal scroll bar can occupy the entire width of the window, or it can be offset by any multiple of fifteen pixels, from 15 to 225. The offset is from the left edge of the window. There are so many options with each of these elements that it's impossible to list all of the features here.



Angel Food Cake
Meringues
Soufflé
Tinsey little wafer

Swiss Mocha Coffee
Double-Fudge Chocolate Cream Pie
Strawberries (dipped in chocolate)

A variety of Group Box styles can be created with one line of code. This window's left side, a standard Mac look, maps to a white background on 4-bit monitors.

Figure 12. Panels and a 3D look

Water's Edge saved the best window for last in its Demo menu; it shows how 3D panels can be used to spruce up your user interface (Figure 12). I like the raised and inset panels best. This figure shows how button groups and lists with a variety of appearances can be created easily by using the Tools Plus libraries. Once again, each of the elements in the figure was created with a single statement in the Demo program's source code; this is clear testimony to the power of the libraries.

There are more features shown in the Demo program than are pictured here. Tools Plus provides floating palettes that are oriented either horizontally or vertically, and also tool bars that nestle directly under the menu bar in your application. All in all, it's an incredibly rich collection of tools.

Apple Computer, Inc. Software License

PLEASE READ THIS SOFTWARE LICENSE AGREEMENT ("LICENSE") CAREFULLY BEFORE USING THE SOFTWARE. BY USING THE SOFTWARE, YOU ARE AGREEING TO BE BOUND BY THE TERMS OF THIS LICENSE. IF YOU DO NOT AGREE TO THE TERMS OF THIS LICENSE, PROMPTLY RETURN THE SOFTWARE TO THE PLACE WHERE YOU OBTAINED IT FOR A REFUND.

1. License. The software accompanying this License (the "Apple Software") is licensed, not sold, to you by Apple Computer, Inc. ("Apple"). Apple and/or Apple's licensor(s) retain title to the Apple Software. The Apple Software accompanying this License and any copies which this License authorizes you to make are subject to this License.

2. Permitted Uses and Restrictions. You may use and install the Apple Software on as many of your Apple-labeled or Apple-licensed computers ("MacOS computers") as are reasonably necessary to develop OpenDoc-compatible parts, containers, and applications designed to operate in combination with MacOS computers ("Applications"). You may not use the Apple Software for any other purpose, including but not limited to, in the development of, or for incorporation into, operating system software. In connection with the development of your Applications, you may also use, incorporate into your own Applications, compile, copy and distribute (in object code form only) as part of your Applications, the Apple Software identified in the Licensing Information folder located on the top level of this CD, provided you reproduce on each copy the copyright information contained on the original copy of the Apple Software. Except as expressly permitted in this License, you may not decompile, reverse engineer, disassemble, modify, rent, lease, loan, sublicense, distribute or create derivative works based upon the Apple Software in whole or part. Your rights under this License will terminate automatically without notice from Apple if you fail to comply with any term(s) of this License.

3. Disclaimer Of Warranty. Some of the Apple Software may be designated as alpha, beta, development, pre-release, untested, or not fully tested versions of the Apple Software. Such Apple Software may contain errors that could cause failures or loss of data, and may be incomplete or contain inaccuracies. You expressly acknowledge and agree that use of the Apple Software is at your sole risk. The Apple Software is provided "AS IS" and without warranty of any kind and Apple and Apple's licensor(s) (for the purposes of Sections 3 and 4, Apple and Apple's licensor(s) shall be collectively referred to as "Apple") EXPRESSLY DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. APPLE DOES NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE APPLE SOFTWARE WILL MEET YOUR REQUIREMENTS, OR THAT THE OPERATION OF THE APPLE SOFTWARE WILL BE UNINTERRUPTED OR ERROR-FREE, OR THAT DEFECTS IN THE APPLE SOFTWARE WILL BE CORRECTED. FURTHERMORE, APPLE DOES NOT WARRANT OR MAKE ANY REPRESENTATIONS REGARDING THE USE OR THE RESULTS OF THE USE OF THE APPLE SOFTWARE OR IN TERMS OF THEIR CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE. NO ORAL OR WRITTEN INFORMATION OR ADVICE GIVEN BY APPLE OR AN APPLE AUTHORIZED REPRESENTATIVE SHALL CREATE A WARRANTY OR IN ANY WAY INCREASE THE SCOPE OF THIS WARRANTY. SHOULD THE APPLE SOFTWARE PROVE DEFECTIVE, YOU (AND NOT APPLE OR AN APPLE AUTHORIZED REPRESENTATIVE) ASSUME THE ENTIRE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION. THE LICENSE FEES FOR THE APPLE SOFTWARE REFLECT THIS ALLOCATION OF RISK. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO THE ABOVE EXCLUSION MAY NOT APPLY TO YOU.

4. Limitation Of Liability. UNDER NO CIRCUMSTANCES INCLUDING NEGLIGENCE, SHALL APPLE BE LIABLE FOR ANY INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES THAT RESULT FROM THE USE OR INABILITY TO USE THE APPLE SOFTWARE, EVEN IF APPLE OR AN APPLE AUTHORIZED REPRESENTATIVE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SOME JURISDICTIONS DO NOT ALLOW THE LIMITATION OR EXCLUSION OF LIABILITY FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES SO THIS LIMITATION MAY NOT APPLY TO YOU. In no event shall Apple's total liability to you for all damages, losses, and causes of action (whether in contract, tort (including negligence) or otherwise) exceed the amount paid for this License.

5. Export Law Assurances. You agree that the Apple Software will not be exported outside the United States except as authorized by United States law. You also agree that Apple Software that has been rightfully obtained outside of the United States shall not be re-exported except as authorized by the laws of the United States and of the jurisdiction in which the Apple Software was obtained.

6. Government End Users. If the Apple Software is supplied to the Department of Defense ("DoD") of the United States Government, the Apple Software is classified as "Commercial Computer Software" and the DoD only acquires "restricted rights" as defined in Clause 252.227-7013(c)(1) of DFARS. If the Apple Software is supplied to any other unit of the United States Government, the Government's rights are as defined in Clause 52.227-19(c)(2) of FAR or, in the case of NASA, as defined in Clause 18-52.227-86(d) of the NASA Supplement to the FAR.

7. Controlling Law and Severability. This License shall be governed by the laws of the United States and the State of California. If for any reason a court of competent jurisdiction finds any provision, or portion thereof, to be unenforceable, the remainder of this License shall continue in full force and effect.

8. Complete Agreement. This License constitutes the entire agreement between the parties with respect to the use of the Apple Software and supersedes all prior or contemporaneous understandings regarding such subject matter. No amendment to or modification of this License will be binding unless in writing and signed by Apple.

Apprentice 4

The Definitive Collection of Source Code and Utilities for Mac Programmers

Over 640 megabytes of high-quality and up-to-date Mac-only source code examples. **All of the source code examples are new and updated in this release!** Apprentice source is mostly C, C++, and Pascal, using CodeWarrior, Symantec, and MPW. Includes examples of applications, games, code resources, control panels, system extensions, plug-in modules, hundreds of snippets, and much more!

$35. Upgrade from any previous Apprentice release for only $25!

Shipping included for U.S. and Canadian orders. Add $5 for shipping outside the U.S. and Canada.

Visa, MasterCard, American Express, and Discover gladly accepted

Celestin Company, Inc., 5652 NE Meadow Road, Kingston, WA 98346
800 835 5514 • 360 297 8091 • 360 297 8092 fax
Internet: celestin@celestin.com • http://www.celestin.com/

The product comes with a 392-page printed spiral-bound manual. It's comprehensive and well written, and includes prototypes (in both C and Pascal) for calling the functions, plus examples and tips. It's really good. And you also have the source code for the Demo application to compare against the manual's explanations. Water's Edge also provides source code tutorial samples illustrating how to create a "Hello World" application, plus buttons and panels, list boxes, editing fields, pop-up menus, picture buttons, tool bars and floating palettes, and so on. Each tutorial project is simple enough that the relevant features are all in one place.

SUPERCDEFS TOO!

Water's Edge also offers a set of custom CDEFs (control definition resources) called SuperCDEFs, implementing many of the custom features of the Tools Plus libraries. As stated earlier, this product is available separately or with the purchase of a Tools Plus Development Kit for any of the listed development environments (not with an academic discount). To paraphrase the *SuperCDEFs User's Manual*, it consists of a collection of high-quality contemporary controls, designed and implemented by professional Macintosh developers to be as reliable as Apple's controls, and just as easy to use. The SuperCDEFs offer a range of styles, from subtle improvements over Apple's own to a bold 3D appearance.

The CDEFs are arranged in categories encompassing buttons, tabs, sliders, and thermometer controls. 68K, native PPC, and fat versions of the controls are offered. The SuperCDEFs product comes with a comprehensive online user's manual. I didn't have a chance to test the CDEFs.

OK, THEY DID IT, BUT CAN I?

I think the answer to this question is: Maybe. It all depends on how much effort you are willing to put into learning the Tools Plus library functions.

One thing you'll have to get used to is that the Tools Plus libraries provide a higher-level event loop called `PollSystem` that returns the current event in a special `Poll` structure. This lets the loop identify directly the element for which the event is destined, rather than requiring you to deal with mouse coordinates, "control parts," or the like. The Tools Plus manual provides only one example of the main event loop, but it is in Pascal, rather than in both C and Pascal. Water's Edge didn't also provide a single library function to initialize the various toolbox managers, so you will have to start your application with the usual toolbox initialization calls. You also have to initialize the Tools Plus components with a call to `InitToolsPlus`. None of this is hard.

More important, there is no graphical interface for designing the interface. You have to design on graph paper, estimating coordinate values for the various elements – because, instead of using resources to describe the various controls, Water's Edge requires that you include coordinates and sizes inside the function calls. This also has implications for "tweaking" the interface as the program design progresses – this involves modifying the code and not just the parameters of a resource.

The folks at Water's Edge are aware of all this, and are currently working on a project code-named "Anthem" that will offer a MacDraw-like user interface design capability, and then generate source code targeted directly at the Tools Plus library function call interfaces. If this sounds like another AppMaker or Visual Architect tool to you, Water's Edge is quick to point out that their objective is to reduce the complexity and volume of source code in a project, and this includes source code in a class library or code generated by an application builder. We'll have to wait until the end of 1996 to see what Water's Edge has in store for us in this regard.

In the meantime, if you are interested in developing applications that have "quality" written all over them, then Tools Plus is for you. It will take some time to get used to writing functions rather than using resources, but the result will be worth the effort. I tested both the CodeWarrior and Symantec versions of the product and found each of them to work perfectly. If I wasn't hard at work on another book project at this time, I'd be very tempted to write a program using the Tools Plus product.

Water's Edge Software, 2441 Lakeshore Road West, Box 70022,
Oakville, Ontario, Canada, L6L 6M9.
Phone: 1-416-219-5628 Fax: 1-905-847-1638
WaterEdgSW@aol.com
http://www.interlog.com/~wateredg

FROM THE FACTORY FLOOR

By Dave Mark

A Little CodeWarrior History

This month, we're going to talk with John McEnerney, one of the compiler writers at Metrowerks.

Dave: How did you first hook up with Metrowerks?

John: I first met Greg Galanos when I was the development manager at Symantec's Language Products Group. Greg was trying to get me interested in doing some sort of deal with the fledgling Metrowerks, and I mostly ignored him because they were trying to compete aggressively with my first product, THINK Pascal. I would have never guessed that a few years later he would offer me the best opportunity of my career.

Dave: When did you leave Symantec?

John: I left Symantec in October '92, taking about 6 months off to figure out what I wanted to do next. I didn't have any real plans, but I figured I'd find some way to do PowerPC work. I didn't relish the thought of trying to write an entire C++ compiler, so I considered doing a Pascal product on my own.

Around this time, Greg had heard from Rich Siegel (of BBEdit fame) that I was no longer at Symantec, and he called me right away. The first thing he said to me was, "describe your dream job," and I told him I wanted to write a PowerPC code generator for the upcoming Power Macintoshes. I flew to Montreal to meet him and his partner, Jean Belanger. We had some Italian food, drank some wine, and they told me a little about their Pascal and Modula products. I was really hot to write a PowerPC backend, but I was not that impressed with their technology. We talked about various contracts, but I didn't have a really solid feeling yet.

Dave: What finally convinced you to go with Metrowerks?

John: In February '93, Greg asked me to meet with him in Palo Alto to get a look at a C compiler that they had just acquired; a guy named Andreas Hommel in Hamburg had been writing it as a hobby. It ran on the Macintosh, had a simple but nice IDE reminiscent of early versions of THINK C, and it was fast. I spent about an hour looking through the source code: it was well organized, the compiler front-end and back-end were cleanly separated, the code was easy to follow, and in addition to being a full ANSI C compiler, it had a lot of the C++ language implemented already.

It was clear that Greg had found a diamond in the rough, the perfect platform for a native PowerMacintosh product. A few hours later we had a contract – I had about 6–8 months to write a PowerPC back-end and linker. Andreas would finish the C++ language implementation, and a few guys in Montreal (Berardino Baratta, Marcel Achim) would work on the IDE and a new Pascal front-end. We immediately hired Greg Dow, who had written the THINK Class Library for Symantec when I was there, to write a new application framework: PowerPlant.

John McEnerney has been writing Macintosh development tools for the past 11 years, first at THINK Technologies / Symantec Corporation, and then at Metrowerks. He was the creator of THINK Pascal, the first Macintosh compiler with source-level debugging, and one of the designers of the THINK C compiler and environment. He is currently the PowerPC compiler and linker architect at Metrowerks.

KILL YOUR BUGS BEFORE THEY KILL YOUR SCHEDULE!
TMON
Accelerated for Power Macintosh
PROFESSIONAL SYMBOLIC DEBUGGER
MindVision Software
www.mindvision.com
tel (402) 477-3269
fax (402) 477-1395

We must have hooked up with Jordan Mattson from Apple around this time, because a week or so later he sent me one of their RS/6000s to help me get started. Between him and Alan Lillich, who I had met at all the early PowerPC meetings that Apple had been holding for their key developers, I got pretty much everything I needed from Apple.

So, I now had a contract to do the most interesting work I could imagine; all I had to do was figure out where to start.

Dave: What was it like working with Andreas' compiler?

John: Andreas' compiler was pretty traditional in its organization. The front-end made a single pass over the source code, performing lexical analysis as it went, and generated an intermediate representation (IR) that consisted of expression trees, labels, and branches. It took about a week to totally remove the 68K code generator from the rest of the compiler, and put in stub routines where the front-end and the back-end connected so that everything would still link. If I could fill in all the stub routines in exactly the right way, we'd have a PowerPC compiler.

The first thing I did was write a routine that dumped the IR in human-readable form – I don't know how Andreas got his 68K code generator to work without that, I guess he can keep more in his head than I can. Looking at the expression trees on the screen allowed me to visualize how the code generator would proceed.

Most CISC compilers spend a lot of time working on the IR trees themselves. Traditional global optimizations like loop-invariant code motion or common subexpression elimination are performed by rewriting the IR trees into optimized IR trees. The code generator gathers information about the shape of the trees, deciding how many registers will be needed, which addressing modes will be used, etc. After instructions are generated they are largely ignored except for small "peephole" optimizations. (A notable exception to this is the gcc compiler, which transforms the expressions into a simple algebraic representation called RTL and uses repeated "peephole" optimizations derived from a machine description to coalesce these RTL expressions into complex instructions and addressing modes.)

Most of the RISC compilers that I'd read about in the compiler literature used a different approach: immediately transform the IR trees into a low-level representation that was similar or identical to the actual RISC instructions of the target machine, and perform all optimizations at the machine instruction level. I decided to use this technique in my PowerPC code generator.

Dave: What was your basic approach to code generation?

John: Strange as it may seem, the first part of the code generator I actually wrote was the instruction scheduler – the phase that reorders instructions to minimize latencies caused by load delays, and to permit floating-point and integer instructions to execute in parallel. I needed to know if my low-level representation – I called it a "pcode" (no relation to the UCSD Pascal pcode) – had enough information for all the phases I would eventually write, and since the scheduler needed a lot of information, it would serve to prove the design of the pcode. Of course, I had to rewrite the scheduler twice more: the first time was to fix the original one, which had some design flaws, and the second time was to make it more general to support 601, 603 and 604 CPUs.

Once I finished the scheduler, I had my data structures organized and all the support routines in place, so I started writing the instruction selection phase – the "guts" of the code generator. This phase visits the IR tree and generates pcode. It does try to recognize certain tree patterns, like opportunities for FMADD and FMSUB routines, but since there are no complex addressing modes and very few complex instructions, it is mostly a straightforward translation to PowerPC instructions.

To get short-term results, I wrote a quick-and-dirty register allocator, and some code to display the generated pcode instructions, and was able to get most of the code generation debugged this way. I decided to use a proprietary object code format, derived from the one we were already using in our 68K compiler and linker, since I could get this working faster than trying to write an XCOFF linker. I spent a few weeks getting the linker working, finished the part of the code generator that wrote the object file, and I could actually compile and link small programs.

Dave: How did the debugger fit into all this?

John: Around August '93 the project was falling into place, but we still didn't have a source-level debugger. In a most serendipitous event, Dan Podwall, a friend of mine from Symantec, called and asked whether there were any opportunities at Metrowerks. Greg Galanos called him right away, hired him on the phone, and 4 weeks later he had written a debugger – in PowerPlant, no less – that could single-step and set breakpoints. This would be the first commercial PowerPlant program – in fact, the first PowerPlant program of any kind aside from Greg Dow's demos.

Dave: How did you build the compiler?

John: By September '93 we had some prototype PowerPC hardware, and I had a working code generator and linker

MENU MILL

is a collection of five menu definition resources (MDEFs) that you just paste into your project's resource file.

Demo
Geneva 9
3D Tools
Colors
ASCII Chart
This Year
Red
Green
Blue
✓Check
◇Diamond
• Bullet
- Dash
Disable
Enable

Build a menu using the MDEF and then just handle menu selections like any normal menu.

Also contains Maitre d', a special menu delevolment tool that assists you in designing your MENU MILL Menus.

MENU MILL

- Use your drawing programs as programming tools
- Fast unobtrusive RGB color picker
- Space saving Geneva 9 MDEF
- Calander Menu

JUST...$69.95

Write, Ariel Publishing, Inc.
11A Leisure Time Drive,
Diamondhead, MS 39525
Call (601) 255-6713
FAX (601) 255-7086

DEVELOPED BY STAZ SOFTWARE, INC

ALL BUGS ARE STUPID.

But spending tedious hours trying to track them down is dumber still. Why not let a tool do the work? QC can find many of those mistakes automatically. Ever write data beyond the end of a memory block? Ever rely on a handle that was purged? Ever call DisposeHandle on a resource or ReleaseResource on a handle? Sure you have! Maybe you just haven't found out about it yet... QC finds these errors and more.

BECAUSE:
Every programmer makes mistakes.
All programs ship with bugs.
Marketing just cut the beta.
You could use some sleep.

YOU NEED:
QC

QC is cool and, unlike other development tools, QC is easy. Try it **FREE:**

1. Connect to our web site
2. Download QC (less than 200K)
3. Send us email to get a serial #
4. Run the installer
5. Run your program
6. Press shift-option-q

"I only have 6 non-Apple Control Panels on my development machine. QC is one of them. 'Nuff said." -Bill Goodman, Compact Pro author

"We wouldn't ship a product without QC's approval." -Mate Gross, Claris Corporation

NOW POWERPC NATIVE! EXISTING USERS UPGRADE FREE!

Onyx Technology

$99

Onyx Technology 7811 27th Ave Bradenton, FL 34209
Tel: 941.795.7801 Fax: 941.795.5901
Web: http://www.std.com/onyxtech/
AOL: OnyxTech AppleLink: D2238 CIS: 70550,1377

Can you spot the difference?

Plenty of people can't. Because whether you update software with a full set of program disks, or a file made with UpdateMaker 2, the result is the same. Guaranteed. UpdateMaker updates are totally ***reliable***. Its system of 32-bit checksums ensures that it updates the right file.

And UpdateMaker is ***easy-to-use*** – simply specify the files and UpdateMaker builds the update. There is no scripting or use of ResEdit. It's even easier for end-users – just one button to press.

UpdateMaker 2 works with any type of Macintosh file. The updater files are extremely ***compact***. And the program options numerous. You can preserve or override user customisations. Save files in Binhex format. Update up to 20 old versions with one file.

The real difference is the savings in time and money. Which explains why many of the best-known names in software development have been using UpdateMaker for years.

Distribution is unrestricted and royalty-free.

Accelerated for Power Macintosh

Only $225, order now by fax: (415) 964 2886

Email: update@adi.com.au

UpdateMaker 2™

ADInstruments, 2225 Grant Road, Suite #4, Los Altos, CA 94024 Phone: (415) 964 2878

which ran on the 68K Macintosh and generated PEF executables that ran on the prototypes. Since this compiler was already built using our own 68K compiler, it was pretty easy to rehost it on the Power Macintosh: we made the changes for the Universal Headers and routine descriptors and such, then compiled it with itself on the 68K machine, which gave us (after some debugging!) a working PowerPC-hosted PowerPC compiler. With a little bit of trickery, mandated by differences between PowerPC floating-point hardware and the 68K SANE software floating-point architecture, we were able to rehost the 68K compiler on the Power Macintosh as well. We now had the fastest compilers on the Macintosh.

Dave: What next?

John: I still had a lot of work to do on the PowerPC code generator. The biggest task was to replace the quick-and-dirty register allocator with a graph coloring–based allocator. This is one of the great algorithms in the history of compilers. For years people had been trying to come up with an accurate way to represent the lifetimes of variables, so that variables or temporaries that did not overlap could share a register. A lot of *ad hoc* techniques were developed, but this guy from IBM Watson Research Center named Greg Chaitin discovered a formal approach that solved the problem better than anything that had been previously attempted: build an "interference graph" which has an edge between any two variables whose values may be live at the same time, and then try to color this graph with N colors where N is the number of available registers.

So my code generator assumes it has an infinite number of "virtual" registers, and generates the most efficient code it can under that assumption; for example, it assumes that all local variables, arguments, and TOC pointers can be assigned to a register. After the code is all generated, the register allocator tries to rewrite the virtual registers using real PowerPC registers, and generates extra code to "spill" values that couldn't get a real register. In most cases, everything gets a register since there are so many on the PowerPC. The smarter register allocator probably makes the overall largest contribution to code quality.

The algorithm has one drawback: it has O(N^2) complexity. There are actually programs which have so many intermediate expressions that the interference graph gets too large and it takes several minutes to color it. So I had to keep around the quick-and-dirty allocator as well, which is why you'll sometimes get an annoying message that says the code generator ran out of registers if you're compiling without global optimizations.

Dave: And so, CodeWarrior was born!

John: By December I had pretty much everything working. After a last-minute dash to get C++ language support working on the PowerPC, we were able to burn our first public release, DR/1, starting a long Metrowerks tradition of getting things in under the wire and never missing a ship date. We introduced the product at the San Francisco Macworld Expo with our huge 8-page *MacWeek* advertisement, and CodeWarrior™ was born.

There were plenty of things to be cleaned up between DR/1 and DR/3, which was our real "1.0" release. But by shipping DR/1 and DR/2 when we did, and by working closely with a lot of the major Macintosh software vendors, we were able to help a lot of companies get their software ported to the PowerMac that otherwise might not have.

For me, I had accomplished what I had wanted to when I was back at Symantec: building the PowerPC compiler that most users would use to port their code to the new Power Macintoshes. And Greg had kept his promise and given me my dream job.

It's about business-to-business, home-to-office, artist-to-client, teacher-to-child.

It's about having the knowledge and the know-how to be the best.

It's about making smart choices today and learning about what you need for tomorrow.

It's MACWORLD Expo.
See the best in Macintosh and complimentary tools and technologies. Explore an exhibit floor with everything you've ever wondered about. . . the internet, networking, databases, education, desktop applications, music, design, publishing, illustration, animation, CAD/CAM, programming, entertainment and multimedia. With a complete conference program featuring over 80 sessions, this is your opportunity to interact with leading experts in the industry and get the knowledge and solutions you've been seeking.

MACWORLD Expo/Boston

MACWORLD Expo/San Francisco

See us on the WWW at:
http://www.mha.com/macworldexpo

For more information call
800-645-EXPO x100

Please send me more information on MACWORLD Expo!
○ **Boston** ○ **San Francisco**
I am interested in: ○ Exhibiting ○ Attending MT

Name________________________________

Title ________________________________

Company________________________________

Address ________________________________

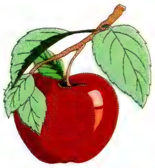
City/State/Zip ________________________________

Phone________________ Fax________________

e-Mail________________________________

Mail to: MHA Event Management, 1400 Providence Hwy., P.O. Box 9103
Norwood MA 02062. Or Fax to: 617-440-0357 Phone: 617-551-9800

CRABB'S APPLE



By Don Crabb

Too Many Cups of Java

In less than a year, Sun Microsystems' Java WWW programming language has gone from "What's that?" to "It will change your life." Companies have been falling all over each other trying to figure out what Java is, how to use it, how to exploit it, how to build products with it. The action's been hot in both the Windows world and in Mac OS land. My question is this: where does the hype end and the reality begin?

For Mac language vendors, the answer is obvious: the reality is that they must have their own version of Java to feed the Mac development community and keep those developers in line with their mainstream C/C++ environments. Besides the problematic SunSoft Java WorkShop 1.0 (problematic for Mac developers, that is, because it runs only on Solaris 2.4 [or later], Microsoft Windows95, and Windows NT 3.51 [or later]), this has already given rise to Natural Intelligence's Roaster, Symantec's Café, and Metrowerks' CodeWarrior Java (thank you, Metrowerks, for not choosing a cutesy name!), and others that will certainly make their presence felt in 1996 – including, maybe, an Apple Java as part of a forthcoming ETO release.

And, as popular as Java-for-the-Mac releases have become, books that purport to teach you Java or some aspect of Java programming have proliferated at an astounding rate. You can walk into any decent bookstore these days and find a Java Programming section all to itself. (By the way, if you are looking for top-quality Java books, consider those published by O'Reilly Books; they're the top titles.)

I don't have a quarrel with the book publishers or tool vendors; it is their job, after all, to provide us with the means to an end. It's the end that I'm not all that clear on.

What, really, is Java going to do for the moribund Macintosh market? Will it give us the means to create insanely great applications that we couldn't create without it? Will it create new markets for our products that did not exist before? Will it convince customers to buy Macs instead of Wintel? Or will it do none of these things?

CREATING JAVA APPLETS

The creation of Java applets has consumed much of the WWW content work for many professional Web publishers for the last six months. And with the availability of Netscape Navigator 2.X for the Mac, Mac folk have a browser that can actually use these applets as they were meant. But has that capability really helped the Mac?

I don't think so. The reason is simple: given the very shaky state of Apple these days, anything that does not contribute to the corporate bottom line (and thus, indirectly, to our individual bottom lines) is a waste of time and money. So far, the general availability of Java on the Mac has not contributed a dime in direct Mac sales. It's made money for tool vendors and book publishers, but it's not made a dent in Apple's financial woes. And it's not likely to make much of a dent, either, unless Apple can figure a way to stake out some aspect of Java that is uniquely Mac (even if it takes marketing conceits to do it).

Don Crabb is a contributing editor and columnist for *MacTech Magazine, MacWEEK, MacUSER, MacHome Journal,* IDG's *Crustacean-At-Large, MacToday, Win95User, ComputerUser, The Chicago Sun-Times Features Syndicate, The Springfield Union-News, PC Magazine,* and about a million other publications. Don welcomes comments at his Internet address: decc@cs.uchicago.edu. You can also check out his WWW home page at http://www.cs.uchicago.edu/~decc/.

So far, that has not happened, and Apple has no announced plans to remedy it. Java has not yet been componentized by Apple to fit its OpenDoc SOM architecture. It has not yet been incorporated into any scheme with Cyberdog. Apple hasn't even mentioned it in the same breath as Copland. Given the latter's continued story of delay, you might think that, if Apple figured Java would be a big part of their Internet strategy, they would announce that Java support would be built into Copland. Even lip service on this issue would be welcomed.

So far, Apple's only told us that it will incorporate Java into the Mac OS, Newton and other technologies. Chief Scientist Larry Tesler said at the recent Internet World conference that the company would make Java "widely available in its products", such as: the Mac OS; Cyberdog, the company's suite of Internet OpenDoc parts; its Internet servers; the Newton OS; and Apple's "media-authoring technologies". But exactly what that means (will it be the full componentization that's needed?) is quite unclear, as are its Mac OS plans.

CREATING NEW MARKETS

I have no doubts that Java will somehow create new markets, just as the Internet and WWW are doing in a more global sense. What I have great doubts about, however, is how any of these emerging markets will help out Apple, the Mac OS, and Macintosh. Given that the current number-one Web browser (Netscape) has become an afterthought on the Mac, and that other WWW publishing, management, and content-creation tools are never going to see Mac versions, the prognosis for Java on the Mac is not swimmingly positive – especially at doing anything to help create new markets for Mac.

This could change given the right moves by Apple, by an Apple-Sun-Netscape merger or technology agreement, or because of increased pressure by Microsoft forcing their foes together to rally around Java as an anti-Microsoft WWW discriminator.

In their order above, I'd rate those possibilties as moderately likely, moderately unlikely, and nearly impossible.

SLIPPING CUSTOMERS THE CONVINCER

Well, this category is pretty much a no-brainer. Java already runs in many forms on Windows platforms (remember, even the SunSoft Java WorkShop 1.0 runs on Windows95 and NT, but not the MacOS). In fact, it's led the way over Mac implementations of the language and over Java-enabled applications on the Mac. So as a Mac vs. Wintel differentiater, Java ain't it.

But that does not mean that Apple could not (with a hell of a lot of development and marketing work, and maybe a partnership or two) produce a compelling Java-on-the-Mac strategy that *would* be an inducement to go Mac – especially if that strategy revolved around Apple as the vendor of choice for the intranet. Given the Mac OS-based client/server products that already play nice on the Web (good stuff like 4th Dimension, Tango/Butler, and others), Apple could make a Java-based intranet strategy fly. But it would mean serious support for existing tools vendors and a serious commitment to developing real Java-based intranet solutions for MacOS customers.

CRUSHER!™

Data Compression Toolkits

The best high performance, portable compression libraries for DOS, Windows, OS/2, Unix, Macintosh, embedded systems, and practically anything else, period.

Robust, 45-Function API
Buffer and File Compression
Portable Archives and Data
Disk Spanning
Encryption
Self-Extracting Executables
On-line Help
Full C Source Code

DOS	$249
Win16	$299
Win32	$299
OS/2	$349
Unix	$349
Macintosh	$349

FREE DEMO

Tel 606-268-1559
Fax 606-266-0726
info@dcmicro.com
http://www.dcmicro.com

DC Micro Development

Call 1-800-775-1073

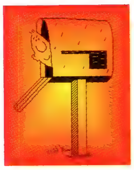
CRABB'S BOTTOM LINE

The question I asked at the top of this column, "What, really, is Java going to do for the moribund Macintosh market?", still remains unanswered. And after slicing and dicing that question into its components, I find the answer just as elusive.

The reason is because the one unknown in all of this is also the most important player – Apple. As I am writing this, Apple still has yet to articulate an Internet strategy. Supposedly CEO Gil Amelio will have told us what that is at WWDC '96 in May. But unless that strategy is a pretty darn bold one that includes some serious commitments (that means people, time, and *real money*) towards redefining Java as an Apple-centric technology driving Apple-branded solutions, I don't think Java's going to mean all that much to Mac developers.

I fervently hope I am wrong, however, and that Apple does try to gain control of the Java bandwagon to meet its and its developers' needs. To date, however, that effort has been pathetic, in the extreme.

DIALOG BOX



By Matt Neuburg, letters@mactech.com

JAVA WITHOUT THE HYPE

Introduction

Talking to my colleagues about Java, I find as many skeptics as enthusiasts. Both sides generally are reacting to the hype, the one rejecting it, the other accepting it. But there are reasons to praise and disparage Java without regard to that hype – that is, without regard to its use on the Internet.

Such a critique is possible because Java is designed to (among other things) support the development of stand-alone, cross-platform applications. So the question at hand really is, "How good is Java as a general-purpose language, and how good is its implementation?"

Java Implementation

As it happens, the rough spots in Java are due mainly to its implementation, which is correctable, not to its language features. The Java libraries are incomplete. I could not, for example, mimic a simple FaceSpan-built interface in Java; it seems not to distinguish checkboxes and radio buttons. The libraries are buggy; list boxes would not reshape or draw correctly, for example. More important, there is no visual interface builder (yet); interfaces have to be created "inside out" – that is, starting with text, as during the Neolithic Age. FaceSpan, Visual Basic, Delphi and others have shown that there should be visual interface builders for a language from day one.

It is also shocking to see the sloppy, unannotated, even unformatted "example" applets made public in the developer's kit.

But implementation problems are correctable; they are significant only until they are fixed. The more important aspects of a language are its feature set and the development methodology that it supports. This is the side of a language that is cast in concrete, and that can make it usable immediately and for the long term – or not. Now the question is, "Does Java represent progress in language design?"

A Little History

Program language design has made great progress during its 50-year history, albeit in steps both forward and backward. Real-life usage constantly suggests or demands new features and, when "enough" demand has accumulated, some brave soul abandons the current languages to design another that addresses at least some of the demands.

The demands upon languages are both concrete and abstract. Concrete demands relate to things like control constructs, data types and fine points of syntax and semantics. More abstract demands are those in support of certain methodologies; these, of course, lead to concrete demands, such as demand for a syntax that supports data abstraction.

It is fairly easy to see the major trends in language design, and to see the progressive and regressive language features. Here are some of the major trends:

- Movement toward "higher-level" control constructs, those that do more in fewer instructions. From assembly-language looping constructs, we moved to the FORTRAN "for" loop, and to the implicit loops of APL or the removal of loops by way of recursion.

- Simplification and reduction, the factoring out of necessary and sufficient features from the current feature experiments. In PL/I, we could loop n times, forward or backward; or loop over any arbitrary list of numbers; or over any combination of forward, backward and arbitrary; and do this "while" or "until" some other condition is true. Pascal reduced this to the simplest "for", "repeat" and "while" loops.

- Putting features into libraries instead of into the language itself. PL/I is a huge language; every conceivable feature (at the time) was put into it. C, on the other hand, is a small core language with large standard libraries.

- Strong typing, which is the detection at compile time of errors that would otherwise occur at run time. Familiar examples of this concern type-checking in expressions and in parameter lists. In FORTRAN and C, actual and formal parameters need not match; in Pascal and Ada, they must match or the program will not compile.

- Information hiding in all its forms, including encapsulation, data abstraction and objects. Opportunities for information hiding within a COBOL program are non-existent; it is very much better in Pascal; and it is key to Object Pascal and C++, both of which support objects.

There was much more in this historical mix – recursion, functional programming, orthogonality, regularity, parameter-passing protocols, and many advances in the implementation of these features – but I think I have highlighted the major ones.

Progress and Regression

The hundreds of languages that have sprung up in the last 50 years, including the dozen that have seen widespread adoption, were not always progressively better, although most fit the moods in which they were designed.

Pascal was progressive because it was a simplification, because it demonstrated the benefits of strong typing, and because it was designed for fast, single-pass compilation. But it was too simple for real-world development until it had good supporting libraries.

C was progressive because it, too, was a simplification, and because it defined a set of libraries as "standard" extensions of the language. But C was also regressive with respect to its arcane declaration syntax, the weakest typing since assembly language, and its exclusion of strings as a native type.

Object Pascal and C++ were progressive because they catenated objects onto compiled languages already in widespread use, but Object Pascal now suffers "socially" from being based upon an abandoned language, while C++ suffers technically from being based upon C.

Java's Place in History

And now Java: How does Java fit into the evolution of languages? It is progressive in several ways, some of which are:

- It is a simplification (but not an over-simplification) of C++, the most popular object-oriented language.
- It reasserts the necessity of strong typing.
- It brings strings back into the language (and makes arrays first-class citizens).
- It provides high-level program-structuring features – packages and interfaces.
- Objects are central to its design; they are not just an add-on or an "orientation".

Java does perpetuate some irritating features of C and C++. For example, it uses the prefix notation for declarations, whereby you sometimes must read a half-dozen qualifiers before you can find an object's name. But, minor irritants aside, Java is a real advance in language design.

This does not mean, of course, that everyone will like its features or even use them to greatest advantage. C hackers (not all C programmers) who now willingly pass along the costs of weak typing and wild-card pointers to their customers will be irritated that there is no switch to turn off responsible programming. And I have already seen a published binary-tree example that simply ignores the object features to implement the tree in a completely "traditional" way. But programming has always included ego, habit, ignorance, self-indulgence and resistance to change, alongside common sense, experimentation, experience, responsibility and progress. Good languages tend to support the good programming traits, but they cannot always suppress the bad.

MOVIE Cleaner Pro
Powerful QuickTime Compression

The Right Tool for Video Compression

If your project includes QuickTime movies, you need Movie Cleaner Pro, the most powerful and easy-to-use video compression tool available. Now you can quickly compress movies for CD-ROM or the Web — with unparalleled quality and speed!

Features include:

Drag & Drop Batch Processing - easily create a batch of up to 2,000 movies in minutes.

Automated IMA audio compression and high quality audio resampling for great sound.

Suspend and Resume - stop compressing and then flawlessly resume - even for batches.

Advanced Settings - easily adjust data rates, A/V fades, high-quality frames, and more.

Adaptive Noise Reduction filter, high quality sine image scaling, and much more!

Terran Interactive
Movie Cleaner Pro 1.2
$189.95*
More info and Orders:
www.Terran-Int.com
1-800-577-3443
International (408) 278-9065

* US Dollars + S&H Source code : MTK 0 496

Summary

Even if we disregard the self-serving hype of all who hope to profit financially from the spread of Java (and especially if we disregard the ignorant parroting of the popular press and the "new world order" freaks), we find that Java is an important step in the evolution of programming languages, a step we all need to take.

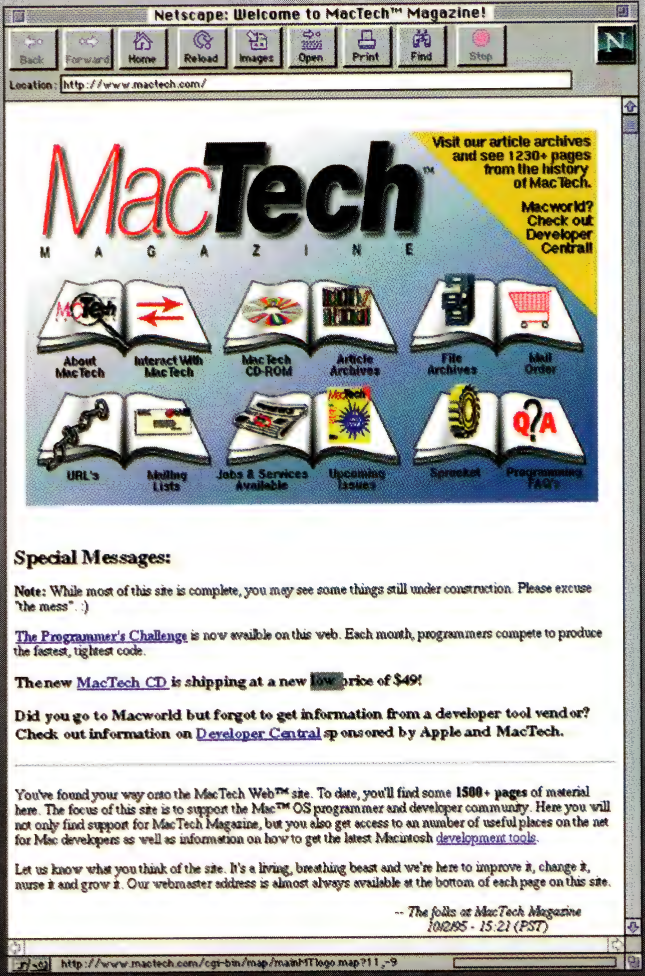
Postscript

Let me make a proposal concerning Java: programmers should hijack it.

Java is an important new tool; we must not accept the idea that it is just for the Internet, or just for "applets", or just for cross-platform development, or that it is just Sun's language. Schools should start teaching it in introductory courses; we should create a demand for good compilers (even if machine-specific) and good development environments; and we should pressure Java's authors and promoters into making the standard libraries correct, comprehensive and truly cross-platform as soon as possible.

– David "Uncle Dave" Moffat, Senior Engineer at SDU
uncle.dave@sdu.com

http://www.mactech.com


Netscape: Welcome to MacTech™ Magazine!
Back
Forward
Home
Reload
Images
Open
Print
Find
Stop
Location: http://www.mactech.com/
MacTech™
MAGAZINE
Visit our article archives and see 1230+ pages from the history of MacTech.
Macworld? Check out Developer Central!
About MacTech
Interact With MacTech
MacTech CD-ROM
Article Archives
File Archives
Mail Order
URL's
Mailing Lists
Jobs & Services Available
Upcoming Issues
Sprocket
Programming FAQ's
http://www.mactech.com/cgi-bin/map/mainMTlogo.map?11,-9

Special Messages:

Note: While most of this site is complete, you may see some things still under construction. Please excuse "the mess". :)

The Programmer's Challenge is now availble on this web. Each month, programmers compete to produce the fastest, tightest code.

The new MacTech CD is shipping at a new low price of $49!

Did you go to Macworld but forgot to get information from a developer tool vendor? Check out information on Developer Central sponsored by Apple and MacTech.

You've found your way onto the MacTech Web™ site. To date, you'll find some **1500+ pages** of material here. The focus of this site is to support the Mac™ OS programmer and developer community. Here you will not only find support for MacTech Magazine, but you also get access to an number of useful places on the net for Mac developers as well as information on how to get the latest Macintosh development tools.

Let us know what you think of the site. It's a living, breathing beast and we're here to improve it, change it, nurse it and grow it. Our webmaster address is almost always available at the bottom of each page on this site.

-- The folks at MacTech Magazine
10/2/95 - 15:21 (PST)

(Need we say more?)

For Macintosh
Programmers & Developers

MacTech™
MAGAZINE

MacRegistry™

Developer Job Opportunities

If you are a Macintosh developer, you should register with us! We have a database that enables us to let you know about job opportunities. When we are asked to do a search by a client company the database is the first place we go. There is no charge for registering. The database service is free. Geographic Coverage is nationwide.

Marketability Assessment - To get a specific feel for your marketability send a resumé via Email or call. You may also request a Resume Workbook & Career Planner.

Discreet - We are very careful to protect the confidentiality of a currently employed developer.

Scientific Placement is managed by graduate engineers, we enjoy a reputation for competent & professional job placement services and we are Mac fanatics.

1-800-231-5920 • das@spi.com • Fax 1-800-757-9003
http://www.scientific.com

Scientific Placement, Inc.

MT, Box 19949, Houston, TX 77224, 713-496-6100 Fax: 713-496-0373
MT, Box 71, San Ramon, CA 94583 510-733-6168 beth@spica.bdt.com
MT, Box 202676, Austin, TX 78720-2676 512-260-0123 lej@zilker.net
AppleLink: D1580; Compuserve: 71250,3001; AOL: davesmall

THE TRATTNER NETWORK

COMPUTER CONSULTING SERVICES

The Trattner Network (TTN) has the best opportunities for Macintosh developers in Northern California and Nationwide.

TTN represents clients with projects in Internet, Multimedia, Codewarrior and OpenDoc development among others.

Now you can visit us our web site at www.tratnet.com and see what openings are available for....

- **Software Developers**
- **QA/QC Professionals**
- **Multimedia Developers**
- **Hardware/Firmware Engineers**
- **Project Coordinator/Managers**
- **Network Professionals**

The Trattner Network has a unique history in **Mac consulting** coupled with exposure to emerging technologies. If you are looking for a chance to enhance your skills and marketability, **please send, fax, e-mail or link your resume to:**

The Trattner Network
Attn: Emily Hoolhorst
170 State Street, Suite 240 • Los Altos, CA 94022
Phone: 415•949•9555 ext.115 ; Fax: 415•949•1026
AppleLink: trat.net ; E-mail: emily@tratnet.com
http://www.tratnet.com

THE DIGITAL TALENT SOURCE

HELP MAKE MACTECH WORK

Here at *MacTech Magazine*, we rely heavily on outside writers for most of the material that appears in our pages. If readers did not participate in the magazine, sending us their ideas and taking the time to write articles, there would be no *MacTech*. *MacTech Magazine* is not a staff of writers sending a constant stream of one-way messages outwards; it's a living, evolving network of readers conversing with one another, educating one another, sharing their knowledge, their experience, their interest, their trials and tribulations and joys and successes in the constantly unfolding story of programming the Macintosh. *MacTech Magazine* doesn't just happen: it's what the community makes it. If we carry reports of future trends and technologies, if we teach useful methods, if we review new books and tools, if we provoke thought, provide help, ride the wave of current interests and concerns, it is only because we reflect the thoughts of our readers, who speak through our pages.

You are invited to involve yourself in this exciting conversation amongst readers. No matter who you are, no matter what your credentials may be, if you have a tale to tell, a trick to share, a technique to teach, we want you to consider joining the family of those who write for *MacTech*.

Don't just wait for a topic to be covered in *MacTech!* Don't just wish some technique would be explained better! Take responsibility! Write us an article yourself!

To write for *MacTech*, just send for our Writer's Kit. It's a Microsoft Word file containing the Styles you need to use, and giving lots of helpful advice and information, including all the legal stuff. You can let us know what you're writing about, if you want suggestions or feedback; this is helpful to us, because it lets us plan for the future. Or if you want to, you can just write the article and spring it on us when it's done. If we publish your article, you'll be paid for it!

Write to me, Matt Neuburg, at editor@mactech.com (or one of the other editorial addresses listed on page 2 of the magazine). Take the future of *MacTech Magazine* into your own hands!

For Macintosh Programmers & Developers
MacTech™
MAGAZINE

NEWSBITS

By John Kawakami

MacTech launches MacDev-1

Westlake Village, CA – May 11, 1996 – Xplain Corporation, publisher of *MacTech™ Magazine*, is proud to announce the launch of MacDev-1™, a new mailing list aimed squarely at the Mac OS Developer Community.

MacDev-1 is a news and information mailing list with a focused charter. Sponsored by *MacTech Magazine* and Developer Depot™, MacDev-1 will be the source of news, information, and occasional special offers specifically for the Mac OS programmer and developer community.

The charter of the MacDev-1 list is to make Mac OS developers more aware of available resources, and to provide an efficient channel through which Mac development tools vendors can spread the word about new product releases. To further this charter, the list is open to press, developers, industry figures, and the general public.

The list will be moderated by *MacTech Magazine*, through a newly appointed Contributing Editor for MacDev-1 – Rich Siegel, the founder, president and CEO of Bare Bones Software, Inc. "MacDev-1 is a natural extension of *MacTech Magazine*'s 'Newsbits' column," said Siegel. "However, readers of the list will hear about the news as it happens. As moderators, our goal is to manage the list traffic, so that subscribers aren't overwhelmed."

Users can subscribe to MacDev-1 by sending mail to macjordomo@listmail.xplain.com with "sub MacDev-1 yourname" in the body of the message. To a submit a posting to MacDev-1, send mail to MacDev-1@listmail.xplain.com.

MacTech Magazine	http://www.mactech.com/
Developer Depot	http://www.devdepot.com/

Apple Looses Flock of Press Releases

Just as the swallows return to Capistrano each year, Apple performed its annual broadcast of the Press Releases during WWDC. This year, over twenty were sent out, and they are all available on the Web. This writer has pored over the text and summarized the interesting news below.

Apple Directions
http://dev.info.apple.com/appledirections/adtoc.html

Apple Propaganda, Straight From the Source

Apple announced plans to expand the company's Web sites, webcasts, and community building activities on the Internet. A new group, Apple Internet Studios and Productions, will manage and coordinate much of the company's Internet presence, which will include not only Apple-specific information, but also general interest content.

Two new Apple-oriented Web sites were announced. The Gil Amelio Homepage – titled "Listen, Learn, & Lead" – outlines Amelio's first-person view of Apple during his first three months at the company, and his strategies for the company's revitalization.

The second site, called "The Plan", outlines the specifics of Apple's recovery plan. People may participate in discussion boards and chats with Apple executives and industry leaders about Apple's developing strategy.

Apple is also doing projects with companies outside the computer industry, including recent promotions for Mission Impossible, and planned webcasts of 1996 Olympic activities.

Gil Amelio Home	http://www.apple.com/leadership/
The Plan	http://ThePlan.apple.com/

Copland is Now Mac OS 8

Apple finally made it official at WWDC; Copland is now known as Mac OS 8. That's "Mac", not "Macintosh", followed by a space, then "OS" and a space, and the number "8". It's not "System 8", and it's not "Copland".

Developer U, Blaze, Debut System 7.5 Course Online

Apple's Developer University is publishing a course produced with a third-party developer. Apple worked with Malcolm Teas of Blaze Technologies, a longtime Macintosh developer, and good explainer.

The printed version of the course can also be ordered from the Developer Tools Catalog.

http://dev.info.apple.com/du/7.5online/7.5TopicsText/ST01-7.5Intro.html

QuickTime VR 2.0 Features New API

Apple has shipped the first developer release of the new programmer's API, and will host developers at a "kitchen" to teach about and get feedback on the new API. Using the API, developers will be able to:

- Display and manipulate panoramas and objects
- Combine still images into panoramas
- Turn QuickTime movies into panoramas
- Combine QuickDraw 3D objects into panoramas
- Incorporate directional sound and speech input
- Incorporate specialized transition behavior between panoramas
- Override QuickTime VR's default behavior relative to event handling, cursors, and hot spots
- Incorporate specialized hot spot information unique to the application
- Manipulate QuickTime VR memory usage

2.0 will also feature "hot spots", or regions where the user may click to follow a URL. QuickTime VR objects will now have zoom and translation capabilities, and also have hot spots. Overall, navigation and interactivity has been improved, and greater integration with QuickDraw 3D is being achieved.

QuickTime VR	http://qtvr.quicktime.apple.com

Apple Lets CyberDog Out, Also OpenDoc DR 5

Bound into this issue of MacTech Magazine are CD ROMs for CyberDog and OpenDoc. As you have probably guessed, these are brand new releases. Have fun!

WEBEDGE IV, SEPTEMBER 16 – 18, 1996, AUSTIN

WebEdge, the conference for Macintosh Web developers, September 16–18, 1996 at Austin's Renaissance Hotel. Topics include: Java for WebMasters, Java Developer Environment, Security, Multimedia tools, introductory and advanced CGI training, advanced HTML and page layout, Databases, Future Technologies, Site Administration Tools and more. The event kicks off with a reception on the 15th. The registration fee for the conference is US $795 if received by June 15, 1996. Thereafter, the fee is $895.

WebEdge • (512) 349-0503

WebEdge	http://www.webedge.com

APPLE AND NATURAL INTELLIGENCE TEAM UP TO BRING JAVA TO THE MAC OS

Apple has committed to supporting Java in the Mac OS, Newton, and Pippin. Apple expects to incorporate Java into its media authoring technologies, Internet servers, and client software, including CyberDog.

To compensate for their limited experience with Java, Apple has licensed Natural Intelligence's Roaster virtual machine, just-in-time compiler, and other technologies for use in Mac OS. Roaster's virtual machine runs Java code, and is distinguished from Sun's original Java virtual machine by its defined API and perfect garbage collection.

Natural Intelligence • (617) 876-7680
Email: info@natural.com

Natural Intelligence	http://www.natural.com/

EVERYWARE'S BUTLER NOW SUPPORTS ODBC, EVERYWARE & NATURAL INTELLIGENCE WORK ON JDBC, TANGO SUPPORTS FILEMAKER

Butler 2.0

Everyware announced that Butler 2.0 now supports ODBC with TCP/IP. With TCP/IP support, Butler now can be accessed by heterogenous clients over the Internet. New enhancements include support for the Alter Table statements, ODBC AppleScript extensions, and support for default ODBC and DAL stored procedures. Included with Butler are the ButlerLink connectivity tools that help developers use Butler as a database back end with C, C++, FirstClass, AppleScript and other OSA languages, and HyperCard via an XCMD. Also included will be a single license for Clear:Access, a visual query and reporting environment that assists end-users with its quick data retrieval and easy-to-use professional report generation capabilities.

Natural Intelligence, Everyware

Everyware and Natural Intelligence announced that they are now working together to create new Java classes to use ODBC over the Internet. They will support the full Java Database Connectivity (JDBC) standard, which allows for Java applications to use ODBC.

Tango With FileMaker

Everyware's Tango is a visual tool for Web developers used to create HTML forms which query databases. Tango started as a tool that generated SQL queries and required an SQL database as the back end. Due to popular demand, Tango now supports interaction with FileMaker Pro databases as well. Web developers can take advantage of FileMaker's rapid application development cycle, and deploy web databases even faster than before.

Everyware Development • (888) 819-2500
Email: sales@everyware.com

Everyware	http://www.everyware.com/
Claris	http://www.claris.com/

For more information on databases on the Macintosh, get a copy of Liam Breck's database development survey:

http://www.cait.wustl.edu/papers/breck.html

A NEW WEBSTAR GETS PLUGGED IN AND CAFFEINATED, WITH HELP FROM NATURAL INTELLIGENCE

The forthcoming WebSTAR 1.3 introduces the WebSTAR Server API (WSAPI), an API that allows developers to write server plug-ins. The plug-ins are similar to CGIs, but will not be restricted by Apple event limitations.

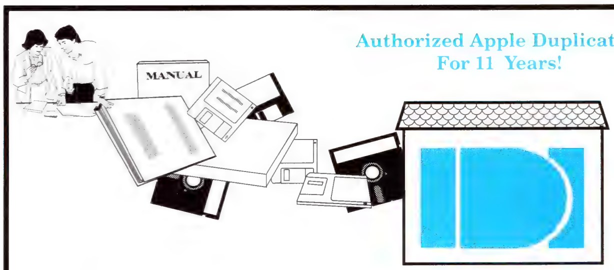
In a separate press release, Natural Intelligence announced that it is working with Quarterdeck to incorporate Java support into a future version of WebSTAR. WebSTAR will be able to execute WSAPI programs written in Java, as well as C, C++ or any other natively compiled language.

Quarterdeck, StarNine division • (510) 649-4949
Email: info@starnine.com

Quarterdeck/StarNine	http://www.starnine.com/

BARE BONES RELEASES BBEDIT 4, SOFTPOLISH 1.2

Bare Bones Software released new versions of BBEdit and SoftPolish. BBEdit, their popular text editor, now features better Frontier support for CGI developers, and code coloring for Java, C, C++, Object Pascal, Rez, 68K assembler, Fortran, Tcl, Perl, HTML, ScriptX, and GuideScript.



Authorized Apple Duplicator
For 11 Years!

From idea to delivery...all handled with top quality service and speed.

Complete Software Duplication Service

FAST

International Datawares is built around things we make: we duplicate software, assemble it, package it. We can handle any part of the job that you want. Or we can handle it all. IDI is equipped to duplicate from Apple to Xerox, for small or large orders, with fast turnaround time. We back up every disk we duplicate. Each is machine-verified bit-by-bit; many are checked by hand. IDI uses top-quality, certified media. What's more, we work with you to provide disk serialization and copy protection schemes to help track sales, register updates, and stop piracy.

We can pack all the pieces that go into your product—disks, manuals, reference cards, and more—and put it all into a shrink-wrapped box. You can provide us with your own packing materials, or use IDI's. Our proven packing materials are sturdy, inexpensive, and will fit in standard packing boxes.

IDI now offers CD-ROM duplication services. Each job is customized so please call us. We'd be happy to fax you the details for your special project.

International Datawares, Inc.
800-222-6032

2278 Trade Zone Blvd • San Jose • CA • 95131
(408)262-6660 fax:(408)262-8906 idikds@aol.com

SoftPolish is a quality assurance tool for Macintosh applications. It checks application resources and identifies potential problems like spelling errors, inconsistent user interface elements, resource inconsistency, and missing resources. Bare Bones has improved SoftPolish to handle more resources, including those generated by modern development environments.

Bare Bones Software • (617) 676-0650
Email: sales@barebones.com
Bare Bones Software http://www.barebones.com/

MAXUM DEVELOPMENT AND AKTIV SOFTWARE RELEASE PHANTOM WEBCRAWLER

Maxum is now the distributor of Phantom, the first commercial Webcrawler for the Macintosh. Webmasters can use Phantom to scan any Web site and build a complete index of its pages.

Once a site has been scanned, Phantom allows users to perform searches using simple queries. The program is "Web server–friendly" and follows all established rules for Web robots, and won't overload servers with simultaneous requests.

"We're thrilled to be working with AKTIV on Phantom," said John O'Fallon, President of Maxum Development. "It represents a tremendous opportunity for Webmasters to improve service to their users, with the ease of use they expect from Mac Web tools. Web crawlers and high-performance searching [are] no longer just for high-end UNIX workstations."

Maxum will be releasing Phantom version 1.1 in early June, but a public beta version will be available May 20th from Maxum's Web site. Phantom 1.1 will be an upgrade to AKTIV Software's "Duppies 1.0" and will be free to all registered users of Duppies.

Maxum Development • (708) 830-1113
Maxum http://www.maxum.com/

EVATAC RELEASES PREDITOR 3.0

Preditor brings together the ease of point-and-click editors and the power of the popular Unix-based editor Emacs. It has a plug-and-play architecture, supports various languages, allows fully styled text, supports scripting, is fully configurable, is integrated with CodeWarrior, and does much more. AppleScript can be edited, recorded and executed all within the Preditor application. The $120 price includes two updates.

Evatac • (703) 620-0669
Email: info@evatac.com
Evatac http://www.evatac.com/

QUED/M 3.0

Who would ever have thought that text editors would become a competitive niche? Recently, QUED/M released a new version, lowered their price to $69, and now comes bundled with Celestin's Apprentice CD. QUED/M 3.0 is similar to the competition, but boasts a powerful grep, noncontiguous and rectangular selection, and a new macro programming language.

Nisus Software • (800) 281-0101
Nisus http://www.nisus-soft.com/

CODEWARRIOR 9

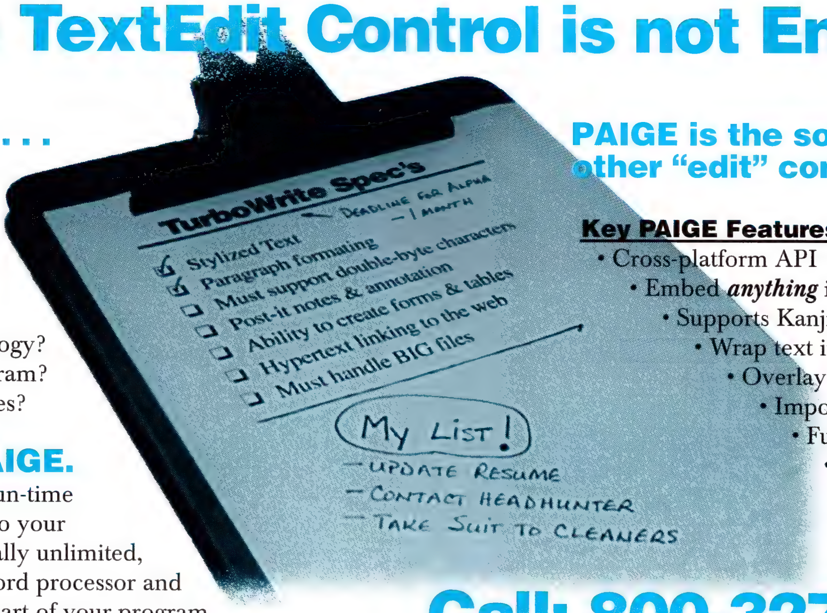
Metrowerks expanded its product line with the release of CodeWarrior 9. Now, in addition to the regular Gold subscription service, they offer a version of CodeWarrior for the BeBox, a Discover Programming Bundle for beginning C/C++ programmers, and another Discover Programming Bundle for Java programmers. There are also the Academic subscriptions for Gold.

Metrowerks http://www.metrowerks.com/

CODEWARRIOR TO SUPPORT ACTIVEX

Who would have guessed that Microsoft's Internet Explorer would turn out to be a Real Macintosh Application? Now Microsoft is getting help from Metrowerks to have Explorer run not only ActiveX controls, but Java applets as well. Furthermore, CodeWarrior will be expanded to include ActiveX development tools. ActiveX controls used to be called OLE controls.

Imagine, on your Macintosh, you can run Java applets, Microsoft ActiveX/COM objects, OpenDoc SOM objects, and CFM code! This writer is reminded of Core Wars, an old


TurboWrite Spec's
Deadline for Alpha — 1 month
☑ Stylized Text
☑ Paragraph formatting
☐ Must support double-byte characters
☐ Post-it notes & annotation
☐ Ability to create forms & tables
☐ Hypertext linking to the web
☐ Must handle BIG files
My List!
— Update Resume
— Contact Headhunter
— Take Suit to Cleaners

For C and C++ Developers ....

When the TextEdit Control is not Enough.

Are you writing a . . .

- Document Viewer?
- Multimedia Product?
- Report Generator?
- Forms Package?
- HTML Editor?
- Email Application?
- Electronic Book Technology?
- Hypertext Enabled Program?
- Program with text features?

Then you need PAIGE.

PAIGE™ is a library of run-time functions you can embed into your application. It will add virtually unlimited, programmable "rich edit" word processor and page layout features to any part of your program.

PAIGE will save you hundreds of precious development hours. Better yet with PAIGE's cross-platform API you can support the most popular desktop operating systems.

PAIGE is the solution when other "edit" controls fail.

Key PAIGE Features

- Cross-platform API
- Embed ***anything*** into text
- Supports Kanji, Chinese, etc
- Wrap text in/around graphics
- Overlay text on graphics
- Import & Export RTF
- Full MFC support
- Royalty Free

Call: 800-327-6703

DataPak Software Inc.

Internet: sales@datapak.com
URL: www.datapak.com/~datapak
CS: 76424,3027
AOL: DATAPAK1
Ph: 360-891-0542
Fx: 360-891-0743

Windows 3.1 • Windows 95 • Windows NT • Macintosh • PowerMacintosh

computer simulation where competing programs battled each other for dominance over address space.

Microsoft's ActiveX documentation
http://www.microsoft.com/intdev/inttech/controls.htm

SMALLTALKAGENTS 2.1

QKS has released an update to SmalltalkAgents. It's available at the ftp site below. Some of the new features are Code Fragment Manager support, PowerPC Accelerations, improved TCP/IP and interface classes, and WASTE text engine support.

SmalltalkAgents 2.1 now supports the Code Fragment Manager (CFM) and CFM-68K. This feature provides direct access to native PowerPC code libraries. Not only can you more easily call code written in other languages, you can also take full advantage of the increasing variety of libraries available from Apple and third parties.

With CFM support, STA is able to deliver third-party code and updaters as DLLs. STA also has an upgrade path where parts of the virtual machine can be extended with accelerated native code fragments.

TextEdit has been replaced with the WorldScript-Aware Styled Text Engine (WASTE). WASTE not only removed the 32K TextEdit limit, but also provides WorldScript awareness, styled text, and embeddable objects. WASTE is implemented as a code fragment.

For more detailed information about product improvements, see their Web site.

Quasar Knowledge Systems • Email: info@qks.com
The update is at ftp://ftp.qks.com//ftp/2.1update/
QKS http://www.qks.com/

PICTORIUS DEBUTS ENTRADA AND FREEWARE NET SERVERS

Pictorius, has released Entrada, a full-featured open database client development tool. Entrada is the new name for their Peregrine database tool. Entrada features a graphical database development environment and full programmability with Prograph.

Integrated into Entrada are the Entrada Net Servers, which allow developers to create various Internet servers using Entrada. The product supports all the popular Internet protocols, including ftp, Web, mail, DNS, OpenTransport, Finger, List, and News. The Net Servers part of the Entrada product is freeware (the entire product is commercial). The Net Servers support CGIs, ACGIs and WebStar Filters, as well as a custom technology called Agents. Together the various parts of the Entrada package allow developers to create Internet/Intranet solutions with one set of tools.

Pictorius Inc. • (902) 492-2880
Pictorius http://www.pictorius.com/

TIPS & TIDBITS

By Steve Sisak

TIP OF THE MONTH

GET VERSION NUMBER STRING

Here's a useful function that gets a version number string, suitable for display in a splash screen or an About box, directly from your `'vers'` resource.

`GetVersNumString` is a handy function that reads an application's `'vers'` resource, constructs the version number, and returns it in a Pascal string. It may not be the most efficient way to do this, but it does the job. This function will not include the third piece of the version number if it equals zero (i.e., print "1.1" not "1.1.0"). The `if` statement controlling this decision is easy to remove; more ambitious programmers might make it an optional parameter. ; -)

I've also included Str255-copy and Str255-concatenation routines to make this tip complete, just in case you don't already have your own available.

Michael Trent

```
/* GetVersNumString
 * Reads 'vers' resource #1 and constructs a string like "\p1.0d3"...
 * If it can't find 'vers' #1, it returns an empty string.
 * NOTE: The application resource file must be the active resource file!
 */
void GetVersNumString(Str255 versStr)
{
        Handle   versHdl;
        long     version;
        unsigned char   ver1,ver2,ver3,relStatus,prerelNum;
        Str255   tmp;

        // clear string;
        versStr[0]=0;
```

[*Actually, you should switch to the application resource file and use Get1Resource() here – sgs*]

```
        // read version no. information
        versHdl = GetResource('vers',1);
        if (!versHdl) {
                return;
        }
        version = *((long *)(*versHdl));
        ReleaseResource(versHdl);

        // Set ver1-3, relStatus, prerelNum from the version info.
        // Note that the first two bytes are in an unusual format.
        ver1 = ((char *)&version)[0];
        ver1 = (((ver1 & 0xF0) >> 4) * 10) + (ver1 & 0x0F);
        ver2 = (((char *)&version)[1] & 0xF0) >> 4;
        ver3 = (((char *)&version)[1] & 0x0F);
        relStatus = ((char *)&version)[2];
        prerelNum = ((char *)&version)[3];

        // Insert v1 and v2 into our version string.
        NumToString((long)ver1,tmp);
        PStringCat(versStr,tmp);
        PStringCat(versStr,"\p.");
        NumToString((long)ver2,tmp);
        PStringCat(versStr,tmp);
        // For convenience, we only print the third number if it is non-zero.
        // If you always want all three numbers, remove the if-statement.
        if (ver3) {
                PStringCat(versStr,"\p.");
                NumToString((long)ver3,tmp);
                PStringCat(versStr,tmp);
        }

        // If the release status is development, alpha, or beta, add a
        // 'd', 'a', or 'b' to our version string.
        switch(relStatus){
        case 0x20:          // development
                PStringCat(versStr,"\pd");
                break;
        case 0x40:          // alpha
                PStringCat(versStr,"\pa");
                break;
        case 0x60:          // beta
                PStringCat(versStr,"\pb");
                break;
        default:
                ;
        }

        // lastly, if we've added a 'd', 'a', or 'b', print the pre-release
        // number at the end.
        if (relStatus != 0x80) {
                NumToString((long)prerelNum,tmp);
                PStringCat(versStr,tmp);
        }
}

/* PStringCopy
 * Copy pascal string a into b.
 */
void PStringCopy(Str255 a, Str255 b)
{
        BlockMoveData(a,b, a[0] + 1);
}

/* PStringCat
 * Concatenate pascal strings a & b, return in a. If there isn't enough
 * room in our array to join both strings, return what we can.
 */
void PStringCat(Str255 a, Str255 b)
{
    short len;

    if ((a[0] + b[0]) > 255)
      len = 255-a[0];
    else
      len = b[0];

    BlockMoveData(&(b[1]),&(a[a[0]+1]),len);
    a[0] += len;
}
```

*Send us your tips or we'll install EvenBetterBusError on your machine! On the other hand, we might just pay you **$25** for each tip we use, or **$50** for Tip of the Month. You can take your award in goods, subscriptions or US$. Make sure any code compiles, and send tips (and where to mail your winnings) to our **new Tips e-mail address** at tips@mactech.com. See p. 2 for our other addresses.*

MacTech™ CD•ROM

Volumes 1-11

With MacTech™ Magazine
articles in THINK Reference™ format.
Includes THINK Reference 2.0

For Macintosh Programmers & Developers
MacTech™ MAGAZINE

Bigger, Better, Faster!

Introducing the new MacTech CD-ROM™ Vol. 1-11. Get all 127 issues of *MacTech Magazine*™ plus a new THINK Reference™ for faster access, all the source code, working applications, demos, frameworks, and more – all in one CD!

- 1420+ articles, from all 127 issues of *MacTech Magazine* (1984 – 1995).
- Improved hypertext, and a new THINK Reference Viewer – for lightning-quick access.
- New super-fast word search.
- 100+ MB of source code – use them in your own applications, with no royalties.
- Full version of THINK Reference 2.0. The original online guide to *Inside Macintosh*, Vols. I-VI.
- 80MB of FrameWorks/SFA archives. The most complete set of FrameWorks archives known.
- Sprocket™! MacTech's Tiny Framework that compiles quickly and supports System 7.5 features.
- The best threads from the Macintosh programmer newsgroups, plus thousands of notes, tips, snippets, and gotchas.
- Popular tools that Macintosh programmers use to increase their productivity.

Available Through Developer DEPOT™

Now Only $49!

List Price: $89

$39 for upgrades from previous versions.

(prices do not include shipping and handling)

http://www.devdepot.com • Phone: 800-MACDEV-1 • 805-494-9797 (outside the U.S. and Canada)
Fax: 805-494-9798 • E-mail: orders@devdepot.com

ADVERTISER/ PRODUCT LIST

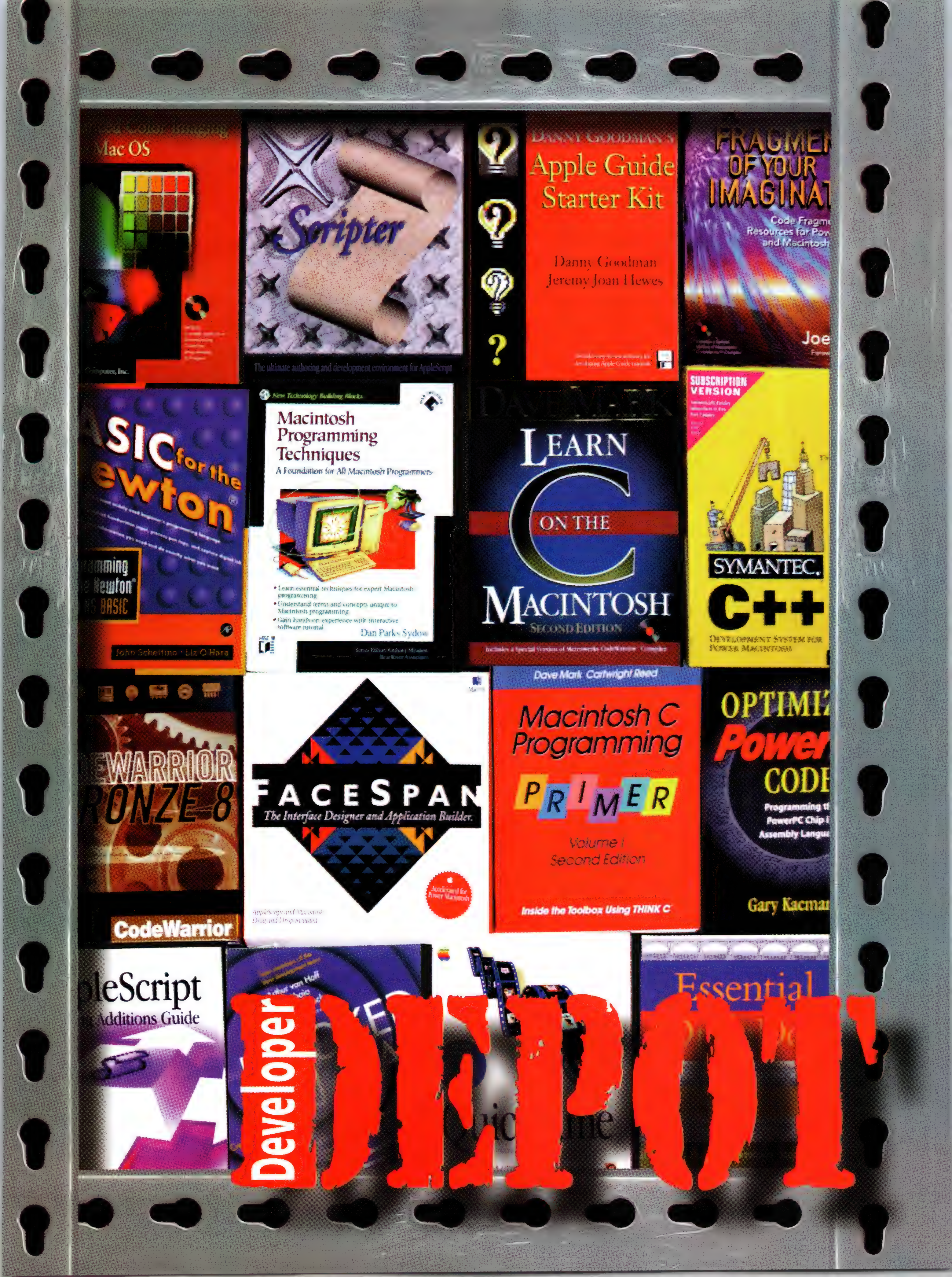
LIST OF ADVERTISERS

A. D. Software 50
Absoft 56
Adianta Inc. 40
ADInstruments 64
Aladdin Knowledge Systems Ltd. 5
Aladdin Systems, Inc. 12-13
Apple Computer, Inc. 57
Apple Developer Relations 18-19
Ariel Publishing 63
Bare Bones Software, Inc. 35
Bowers Development 11
BroadCast Software Corp. 43
Celestin Company, Inc. 60
Cyclos 47
Datapak Software Inc. 75
DC Micro Development 67
Developer Depot 79
Digitool 53
dtF Americas, Inc. 39
Evatac Software 34
FGM, Inc. 37
International Datawares, Inc. 74
Jasik Designs 45
JointSolutions Marketing 22
JYACC ....IBC
MacTech CD-ROM 1-11 77
MacTech Web™ Site 70
Macworld Exposition 65
Mango Tree Software, Inc. 47
Mathemaesthetics, Inc. 1
Metrowerks ....BC
Micro Macro 9
Microsoft 7
MindVision Software 41, 62
Natural Intelligence 20
Neologic Systems 17
Nisus Software, Inc. 38
Now Software 54
Onyx Technology 63
OPtions Computer Consulting 34
Ray Sauers Associates 55
Richey Software Training 44
Scientific Placement 71
Seapine Software, Inc. 32
Symantec ....IFC
Terran Interactive 69
The Trattner Network 71
Water's Edge Software 42
Willows Software Inc. 14

LIST OF PRODUCTS

Apple World Wide Developer Web Site • Apple Computer, Inc. 18-19
AppMaker • Bowers Development 11
Apprentice 4 • Celestin Company, Inc. 60
BBEdit • Bare Bones Software, Inc. 35
BroadCast™ • BroadCast Software Corp. 43
CodeWarrior™ • Metrowerks ....BC
Crusher!™ Data Compression Tool Kits • DC Micro Development 67
Cyberdog CD-ROM • Apple Computer, Inc. 57
The Debugger V2 • Jasik Designs 45
DragInstall 2.0 • Ray Sauers Associates 55
dtF, The Relational Database System • dtF Americas, Inc. 39
Duplication Service • International Datawares, Inc. 74
Fortran 77 for Macintosh • Absoft 56
4D Tool Kit • OPtions Computer Consulting 34
Installer VISE 4.0 • MindVision Software 41
JAM® • JYACC ....IBC
MacApp Programmers • MacXperts 71
MacHASP • Aladdin Knowledge Systems Ltd. 5
Macintosh Developer Tools & Publications • Developer Depot 79
MacNosy • Jasik Designs 45
MacRegistry™ • Scientific Placement 71
MCL • Digitool 53
The Memory Mine™ • Adianta Inc. 40
MENUMILL • Ariel Publishing, Inc. 63
MicroGuard Plus™ • Micro Macro 9
Movie Cleaner Pro™ • Terran Interactive 69
neoAccess™ • Neologic Systems 17
neoShare™ • Neologic Systems 17
Now Utilities™ • Now Software 54
OOFile™ • A.D. Software 50
OpenDialog™ • FGM, Inc. 37
OpenDoc CD-ROM • Apple Computer, Inc. 59
PAIGE • DataPak Software, Inc. 75
Preditor • Evatac Software 24
Programmer Training Courses • Richey Software Training 44
QC • Onyx Technology 63
QUED/M™ 3.0 • Nisus Software, Inc. 38
Recruitment • Scientific Placement 71
Recruitment • The Trattner Network 71
Resorcerer® 1.2 • Mathemaesthetics, Inc. 1
Roaster™ • Natural Intelligence 20
Smaller Installer • Cyclos 47
StuffIt InstallerMaker 3.0 • Aladdin Systems, Inc. 12-13
Symantec C++ • Symantec ....IFC
TCP/IP Scripting Addition • Mango Tree Software, Inc. 47
TestTrack™ • Seapine Software, Inc. 32
TMon Professional • Mindvision Software 62
Tools Plus™ 3.0 • Water's Edge Software 42
Trade Show • Macworld Exposition 65
Twin APIN • Willows Software Inc. 14
UpdateMaker 2™ • ADInstruments 64
Updater VISE 1.1 • MindVision Software 41
Visual FoxPro • Microsoft 7
World Wide Web Pages Creative • JointSolutions Marketing 22

The index on this page is provided as a service to our readers. The publisher does not assume any liability for errors or omissions.


Mac OS
Scripter
The ultimate authoring and development environment for AppleScript
Danny Goodman's
Apple Guide Starter Kit
Danny Goodman
Jeremy Joan Hewes
Fragment of Your Imaginat
Joe
Basic for the Newton
John Schettino · Liz O'Hara
New Technology Building Blocks
Macintosh Programming Techniques
A Foundation for All Macintosh Programmers
Dan Parks Sydow
Dave Mark
Learn C on the Macintosh
Second Edition
Subscription Version
Symantec.
C++
Development System for Power Macintosh
CodeWarrior
Bronze 8
FaceSpan
The Interface Designer and Application Builder.
Dave Mark Cartwright Reed
Macintosh C Programming Primer
Volume I
Second Edition
Inside the Toolbox Using THINK C
Optimiz
Power
Code
Programming the PowerPC Chip in Assembly Language
Gary Kacmar
AppleScript
Additions Guide
QuickTime
Essential

Developer DEPOT

Open Programmer's Hours...

...and all day too. In fact, we're open 24 hours a day, 365 days a year!

Which means that whenever you get a craving for the latest in developer tools, we're just a couple of keystrokes away.

You can browse for hours (we don't mind), pick up the items you want, and conveniently pay with your major credit card at our virtual checkout. Then, kick back and wait for your goodies to arrive - all without ever leaving your house!

Best of all you'll shop confidently, because at Developer Depot, you're guaranteed complete satisfaction and the best prices for 30 days after your purchase.

Talk about a safety net!

That's right, and speaking of nets, get connected and come check out the vastest on-line selection of developer products... any day, any time!

Developer DEPOT
http://www.devdepot.com

Dear Developer,

Since March, 1985, **MacTech™** Magazine (then MacTutor) has been the place for you to find Macintosh Development products – the majors as well as those harder to find. Back then … and today, our focus has always been for the Macintosh developer community to grow and prosper.

But you deserve more than simple listings – and you've asked to make it easier to find and order products while maintaining our renowned customer service and pricing. We've been listening …

We are therefore proud to announce **Developer Depot™** – a new mail order concept and entity dedicated to you, the Mac OS Developer. What makes Developer Depot so good?

- World renowned customer service
- 24-hour and full accessibility via our continually updated web site (http://www.devdepot.com)
- Satisfaction and best price guaranteed
- A new, easy to remember toll free phone number (**800-MACDEV-1**)
- Great selection (hundreds of products)
- Convenience: you can order in many ways (web site, e-mail, fax, phone)
- A new full color, organized, and expanded catalog – updated monthly

Additionally, we are a Macintosh house using an all Macintosh system. Our knowledgeable staff can answer your questions about products – and if they can't, they'll find the answers for you.

While Developer Depot is completely separate from MacTech Magazine, Developer Depot is the primary source for all MacTech products.

Let us know what you think – we're here to serve you.
Developer Depot: The products you need with the service and prices you deserve.

Neil Ticktin
Chief Executive Officer

Andrea J. Sniderman
Chief Operating Officer

TABLE OF CONTENTS

02 MACTECH
03 DEVELOPMENT ENVIRONMENTS
06 INTERNET RELATED
07 SCRIPTING
08 TOOLS, LIBS & UTILITIES
15 BOOKS & REFERENCE

Lowest price guaranteed.
Feel Good about your purchase!
30-day return policy.

One & 2-Day SHIPPING
(call for details)

Order Toll-free
800-MACDEV-1
(800-622-3381)

Developer Depot 30 day Money Back, Price and Satisfaction Guarantee
Developer Depot products are sold with a 30 Day money back guarantee on user satisfaction, lower prices and against defects. If, for any reason, you are not satisfied or find the same product at a lower price within 30 days, please call Customer Service at 800-MACDEV-1 and request a Return Merchandise Authorization (RMA) number to get a full refund or the difference in price (where applicable). You must return undamaged product at your expense, including all its original packaging, documentation and the blank warranty card if applicable. Developer Depot will replace defective product upon receipt of the defective merchandise. Please remember to back up your data before installation of any new hardware, software, or peripherals; we cannot be responsible for any lost data. Policies, item availability, and prices are subject to change without notice. The price in effect when we receive your order will be the price that you are charged. We are not responsible for any typographical errors in this or any other catalog, nor for any misstatements from any vendor. Purchase orders are also accepted but must be in writing, signed, come with a contact person and a telephone number, and mailed to P.O. Box 5200, Westlake Village, CA 91359-5200. Faxed copies and purchase order numbers alone are not acceptable.

Stuff our lawyer made us write.
Developer Depot makes no other warranties. All other warranties, either expressed or implied, including the implied warranty of merchantability and fitness for a particular purpose are disclaimed. Developer Depot shall not be liable for any direct, special, incidental or consequential damages including lost profits, from any delay in delivery, or for any personal injury arising from the use of any product sold through Developer Depot. The limit of direct damages, if any, shall not exceed the purchase price of the product.

© 1996 Xplain Corporation. All rights reserved. Any unauthorized duplication is in violation of federal laws. Developer Depot is a registered trademark of Xplain Corporation. All product names in this catalog are the trademarks of their respective holders.

For Macintosh Programmers & Developers

MacTech™ MAGAZINE

MacTech Magazine

Subscriptions: US magazine: **$47** for 12 issues
Canadian: **$59** for 12 issues
International: **$97** for 12 issues
Back Issues: **$5** each (subject to availability)

MacTech CD-ROM Volumes 1-11 NEW!

- 1420+ articles, from all 127 issues of MacTech Magazine (1984-1995).
- Improved hypertext, and a new THINK Reference Viewer — for lightning quick access!
- New hyperlinks between articles allow you to jump to other relevant articles.
- 100+ MB of source code — use them in your own applications, with no royalties!
- Full version of THINK Reference™ 2.0. The original online guide to Inside Macintosh, Vols. I-VI.
- 80MB of FrameWorks/SFA archives. The most complete set of FrameWorks archives known.
- Sprocket™! MacTech's Tiny Framework that compiles quickly and supports System 7.5 features.
- The best threads from the Macintosh programmer newsgroups plus thousands of notes, tips, snippets, and gotchas.
- Popular tools that Macintosh programmers use to increase their productivity and much more!

List $89.00 Our Price **$49.00** Upgrades: Our Price **$39.00**

MacTech Mouse Pad

Slide on this! With an extra-large surface (11" by 10") and a deluxe sleek plastic coating, you'll be zooming across your screen in no time at all.
Speed limit not enforced!

Our Price **$8.95**

Order Toll-free
800-MACDEV-1
(800-622-3381)

EXCLUSIVES!

FrameWorks Magazine

$8 per back-issue, subject to availability.

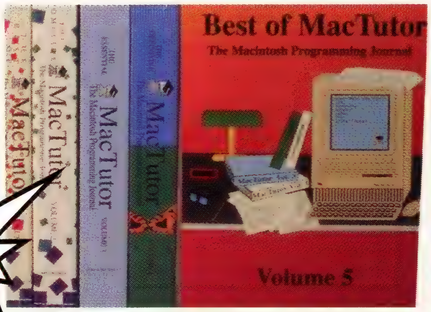
FrameWorks Source Code Disk

$10 per back issue, subject to availability.

MADACON '93 CD-ROM

The highlights of MADACON '93, including Mike Potel on Pink, Bedrock, MacApp, OODLs, and more. Slides, articles, demos, audio, and QuickTime. List $95.00 Our Price **$9.95** per set.

BLOW OUT!



Best of MacTutor

The Best of MacTutor Collection, Vols.3-5.
List $69.00 Our Price **$9.95** per set.

E RELATED PRODUCTS: **Code Manager**

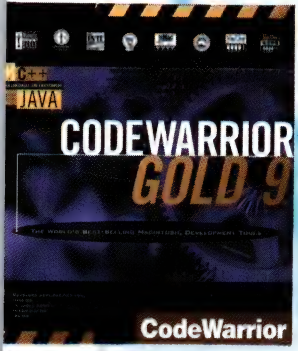
SEE RELATED PRODUCTS: **Inside Power Plant**

SEE RELATED PRODUCTS: **Inside CodeWarrior 8**

SEE RELATED PRODUCTS: **C++ Programming with CodeWarrior**

SEE RELATED PRODUCTS: **PowerMac Programming Starter Kit**

SEE RELATED PRODUCTS: **Discover Programming For Macintosh**



CodeWarrior 9 Gold by Metrowerks

SEE RELATED CATEGORY: **Internet Related**

- Build applications for the Mac OS, Windows 95, Windows NT, Magic Cap, & the new Be OS.
- Includes C, C++ and Object Pascal compilers, source-level debuggers, object-oriented frameworks (PowerPlant, MacApp, MFC), Apple's MPW, complete online documentation and source code examples for all languages and platforms.
- The IDE software has been localized in eight languages plus English. CodeWarrior is sold on an annual subscription basis and has three major releases each year.
- **Includes a 1 year MacTech subscription.**

Our Price **$399.00**

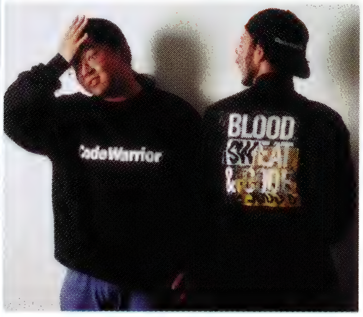
SEE RELATED PRODUCTS: **Learn C on the MacIntosh**

SEE RELATED PRODUCTS: **Metrowerks CodeWarrior Programming**

CodeWarrior 8 Bronze

- Build applications for the Mac OS on 68K machines.
- Includes C, C++ and Object Pascal compilers, source-level debuggers, object-oriented frameworks (PowerPlant, MacApp), Apple's MPW, complete onlinedocumentation and source code examples for all languages and platforms.
- The IDE software has been localized in eight languages plus English.
- Sold on an annual subscription basis and has three major releases each year.
- **Includes a 6 month MacTech subscription.**

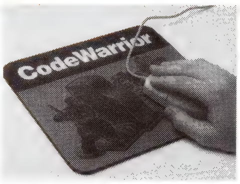
Our Price **$149.00**



CodeWarrior Wear (XL only)

You live it, you breath it... you might as well wear it!

- Black CodeWarrior Sweatshirt: Our Price **$29.95**
- "Blood, Sweat & Code" black long-sleeve shirt: Our Price **$7.95**
- Cross Platform white shirt (not pictured): Our Price **$14.95**
- Hawaii Five-0 shirt (not pictured): Our Price **$7.95**
- Arnold at work T-shirt (not pictured): Our Price **$9.95**
- Hat (black or white): Our Price **$14.95**
- Winter Hat (not pictured): Our Price **$14.95**

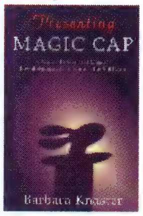


CodeWarrior Mouse Pads

Fight your way through the night with these cool-looking pads!

- Arnold: Our Price **$8.95**
- Gears (not pictured): Our Price **$9.95**

Presenting Magic Cap, A Guide to General Magic's Revolutionary Communicator Software



- Perfect for novices as well as experts who want to take full advantage of Magic Cap.
- Step through its screens, see how it works, and learn ways to use it.
- Describes the power of smart messaging to customize the way you handle e- mail, a name card file that learns how to keep track of your contacts, a datebook that performs like a faithful assistant, and countless other abilities.

List $16.95 Our Price **$15.25**

SmalltalkAgents® by Quasar Knowledge Systems

• Features a new generation of the Smalltalk language, QKS Smalltalk™. • Productivity is no longer measured in lines of code, but in project completion time. • Allows "live" direct manipulation of your objects. • Provides a dynamic, interactive and iterative development process. • Provides easy and full access to the features of the Mac OS™ and Mac Toolbox. • Link your non-Smalltalk code resources using our External Code Linking Toolkit™ (ECLT). • A sophisticated database for source code management provides an almost infinite variety of ways to cross-reference, access, view, and manipulate your code and objects. • Includes an Application Delivery Toolkit™ (ADT) that allows you to create royalty-free, stand-alone, double-clickable applications in just a matter of minutes. • Any component built in AO/S will function as an OpenDoc component or container and components from non-AO/S sources can be seamlessly integrated into the AO/S system.

Our Price **$695.00**

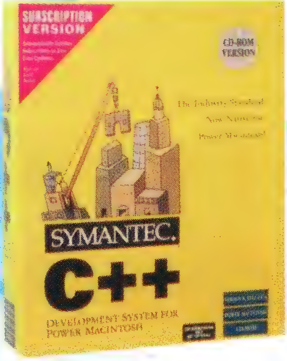
SEE RELATED CATEGORY: **Internet Related**

SEE RELATED PRODUCTS: **Symantec C++ Programming**
SEE RELATED PRODUCTS: **Learn C++ on the Macintosh**
SEE RELATED PRODUCTS: **Programming in Symantec C++**
SEE RELATED PRODUCTS: **Mastering theTHINK Class Library**

Symantec C++ 8.4

- Native for Power Macintosh.
- Develop full-featured Power Macintosh applications quickly and easily!
- Environment includes: a true native Power Mac implementation of the C++ language, MrC/MrC++ compilers for fast Power Mac executable code, Visual Architect for fast, easy GUI generation, a new THINK project management system that supports complex applications, a new editor/browser , a powerful, easy-to-use source code debugger.
- Also includes: The industry-standard THINK Class Library!
- Next two updates free when you send in your registration card and more!

List $499.00 Our Price **$349.00**



Symantec C++ for 68k

- The standard in development languages. • Powerful combination of fully integrated visual tools and the latest in C and C++ compile technology. • Provides an object-oriented approach to application development. • Write code that is extensible, reliable, and maintainable. Includes: Integrated Environment with full source-code debugging, Incremental Linker which eliminates long link times THINK Class Library, AppleEvents, Visual Architect™, THINK Inspector, Project Models, Source-Code Control with integration with Apple's SourceServer (included), Support for Scripting, Powerful Standard Libraries which includes I/O Streams, ANSI standard C library, and sample programs. • Full source code is included – Create applications that run on 68K Macs and Power Macs (emulatec These applications can easily be migrated to native Power Mac, by trading up to Symantec C++ for Power Mac.

List $369.00 Our Price **$99.00**

Think Pascal Version 4.0

by Symantec Corporation

- Great for professionals and students
- Fully integrated for rapid turnaround time – take advantage of System 7 capabilities
- Supports large projects, enhanced THINK Class Library, System 7 compatibility, superior code generation, and smart linking.
- Includes four Macintosh disks, a user manual, and an object-oriented programming manual.

Our Price **$169.00**

LS Object Pascal CD-ROM **by Fortner Research.**

- Includes the world's first Object Pascal compiler for Power Macintos
- 100% compatible with Apple's MPW Pascal
- Combines the best of Apple's native development tools with innovative new technology developed at Language Systems
- Compiler options specify 68K or native PowerPC code generation
- CD includes: LS Object Pascal compiler, Universal Pascal Toolbox interfaces, fully loaded MPW 3.3.1, 68K and PowerPC source debuggers, PowerPC assembler, online documentation, Macintosh Tech Notes, and a special version of AppMaker by Bowers Development that generates native Pascal source code.

List $399.00 Our Price **$349.95**

MachTen – Power UNIX **by Tenon Intersystems**

- Dynamically linked shared libraries, memory mapped file access and integrated UNIX and Macintosh development tools.
- BSD 4.4 and conforms to the Federal Information Processing Standard 151-2 (the POSIX FIPS).
- Pre-emptive multitasking for UNIX applications and includes a full featured high-performance TCP/IP protocol stack that supports multi-homing and multi-casting, features not yet available even with Apple's new Open Transport.
- A complete UNIX software development environment with a source-level debugger and C, C++, and Fortran compilers all generating native PPC code.
- Also included is a high-performance X server and complete X11R5 X.

List $1,045.00 Our Price **$695.00**

SEE RELATED CATEGORY: **Internet Related**

Personal MachTen for 68K Macs

- MachTen UNIX for Macintosh (from Classic on up, including PowerBooks and Duos)
- A Mach-based Berkeley UNIX with pre-emptive multi-tasking
- Extends the Mac OS with UNIX networking and software development tools.
- Built-in internet services include domain name service, POP mail service, internet routing, SLIP & PPP, and Web service.

Our Price **$495.00** Also Available: MachTen XWindows Our Price **$350.00**

SEE RELATED CATEGORY: **Internet Related**

Mjølner BETA System

• Software development environment supporting object-oriented programming in the BETA programming language. • Uniquely expressive and orthogonal. • Unifies just about every abstraction mechanism – including class, procedure, function, coroutine, process and exception. • Includes: general block structure, strong typing, whole/part objects. The compiler: binary code generation, automatic garbage collection, separate compilation, interface to C, Pascal, and assembler. • The system: persistent objects, basic libraries with containers classes, platform-independent GUI application frameworks on Unix, Mac and Windows NT, metaprogramming system. • The Mjølner BETA System for Macintosh requires MPW (basic set) 3.2 or later. Package containing compiler, basic libraries, persistent store, GUI framework, and comprehensive documentation. (Other packages are also available).

List $295.00 Our Price **$249.99**

LS Fortran by Fortner Research

• 68K or PowerMac LS FORTRAN comes in two versions: one that generates 68K code (with or without FPU) and one that generates native code for the PowerMac • Source code is compatible between these two compilers • Programs originally developed with the 68K version can boost execution speed by 4 to 6 times by running the same program on a PowerMac • Supports all VAX intrinsic functions and data types, and virtually all VAX extensions • Built-in Debugging • Free Runtime Licenses • Full-featured editor, multiple window capabilities, and a powerful scripting language • Highly optimized, full ANSI standard, 32-bit clean, FORTRAN 77 compiler • Fully compatible with virtual memory under System 7 • Supports a variety of third-party tools to customize any FORTRAN environment • Power Tools include graphing software, subroutine libraries, and user interface tools • Free technical support.

Our Price **$695.00**

Fortran 77SDK by Absoft

For Power Macintosh includes a globally optimizing native compiler and linker, native Fx™ multi-language debugger, and Apple's MPW development environment.

• The compiler is a full ANSI/ISO Fortran 77 implementation, and includes all MIL-STD 1753 extensions, Cray/Sun-style POINTER, and several Fortran 90 enhancements • MRWE, Absoft's framework library is included in the MIG graphics library • Supports the native Macintosh PPC toolbox • Includes Absoft's Fx debugger which can debug intermixed FORTRAN 77, C, C++, and PPC assembler • The linker compiler, and debugger all run as native PPC tools, and produce Macintosh PPC executables

Our Price **$699.00**

BASIC for the Newton

- A fully interactive implementation of the BASIC programming language.
- Runs entirely on the Newton – no host is required.
- Create files, access the built-in soups, and the serial port for input and output.
- Work directly on the Newton, or through a connected Mac/PC and keyboard.

Our Price **$99.00**

LPA MacProlog

- Comprises a Edinburgh syntax Prolog compiler system set in an attractive multi-window development environment with an integrated program editor, graphical call-graph facilities and an interactive source-level debugger.
- Features high-level access to the Macintosh ToolBox for using graphics, dialogs, windows, icons, resources in a simple and versatile way.
- Includes interfaces to C and Pascal code resources.
- Enables the production of double-clickable distributable applications.
- Optional add-ons tools include flex, Prolog++, MacDBI for Oracle and the MacProlog Dialog Editor.

Programmer Edition **$745.00** Developer Edition (includes the run-time generator and distribution license) **$1500.00**

MacFortran II V 3.3 by Absoft

MacFortran II is a VAX/VMS compatible full ANSI/ISO Fortran 77 compiler including all MIL-STD 1753 extensions

- Comes with the latest version of MPW, and includes SourceBug, and SoftwareFPU
- Includes Absoft's Macintosh runtime Window Environment (MRWE) application framework, and MIG graphics library
- MacFortran II features improved 68040CPU support and is fully compatible with Power Macintosh under emulation

List $595.00 Our Price **$549.99**

SEE RELATED PRODUCTS:

Programming for the Newton using NewtonScript

SEE RELATED PRODUCTS:

Programming for the Newton using NS Basic

Order Toll-free
800-MACDEV-1
(800-622-3381)

Roaster DR2™

From Natural Intelligence Inc. Get the most out of Sun's new Java™ Programming Language!

SEE RELATED CATEGORY: **Dev. Environments**

- Write, test and run Java™ applets on the Macintosh in a full-featured Mac development environment
- Features include: project window that includes a finder-like view of packages, Macintosh native compiler, source code editor with powerful search features and intuitive use interface, runtime engine for quick and easy applet testing.
- Requirements : PowerPC, CD-ROM.

List $399.00 Our Price **$299.00**

TCP/IP Scripting Addition™

SEE RELATED CATEGORY: **Scripting**

- Award-winning AppleScript scripting addition
- Allows you to write scripts using MacTCP™ commands in AppleScript™.
- Send e-mail or files through a script, check if users are logged on (via Finger), automate FTP, Gopher, NetNews, Telnet, and LPR, verify links in HTML documents, and quickly write many other TCP/IP client-server programs.
- Works with AppleScript 1.0 or later and MacTCP 2.0.4 or later. Compatible with Open Transport™ 1.1.

Our Price **$49.00**

SEE RELATED CATEGORY: **Tools, Libs & Utilities**

OOFILE HTML Writer

The first OODBMS framework to offer a complete solution for application authors. Replaceable backend database, currently Faircom's c-tree Plus for cross-platform royalty-free power. PowerPlant and other frameworks integrated with edit fields, database browsers and more. AppMaker users, generate complete applications. User-friendly syntax makes it easier to migrate from FoxPro and the non-OO world. Demo's on CodeWarrior and AppMaker CD's. **$900.00** for a once-off c-tree bundle, or **$1,095.00** 1year subscription. HTML and character-mode report-writer **$195.00**. Full GUI report-writer, including HTML, is **$495.00**. Mac Platform Bundle - includes all Mac frameworks, c-tree, and Report-Writer. **$1,495.00**

Wanna see your product move?

Get it in Developer Depot!

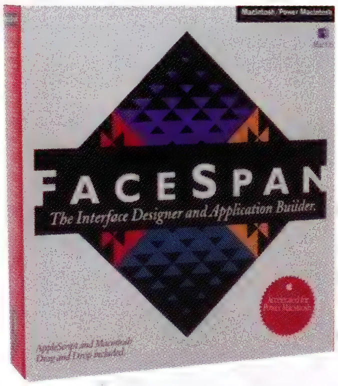
The fastest, cost effective way to get product off your shelves.

For information: • Voice: 805/494-9797 • Fax: 805/494-9798 • E-mail: marketing@devdepot.com

DataScript

• Make your integrated AppleScript solutions database aware-today. • Takes just six lines of AppleScript to make new or existing scripted solutions database aware, and fetch data from RDBM's such as Oracle, Sybase, DB2, or Informix. • Easy to learn, easy to use, and easy to remember. "Inside DataScript" contains lots of easy to follow scripts to reuse in your own solution. • Very attractive licensing schemes.

List $249.00 Our Price **$229.99**



FaceSpan™ v2

Build applications quickly and easily!

- Extensible Rapid Application Designer (RAD).
- Combines an interactive, visual interface design environment with the object-oriented power of AppleScript or any OSA language.
- Integrate the capability of scriptable programs into your custom application.
- Display scrolling lists, popup menus, scrolling text, movies, multi-column tables, pictures, icons, buttons, and others.
- Program custom objects using Pascal or C.
- Includes a royalty-free distribution license, for unlimited runtime users, of your FaceSpan-based applications. Plus a FREE UPGRADE to the next version for registered users!

List $199.00 Our Price **$179.00**

Script Debugger by Late Night Software Ltd.

- A powerful and flexible AppleScript authoring tool – get the most from AppleScript!
- Advanced debugging environment offers single-step script execution with breakpoints.
- Script Debugger dictionary browser features a graphical view of objects provided by scriptable applications.
- Includes Late Night Software Scripting Additions – a collection of more than 70 new AppleScript commands, and Scheduler, a utility that allows you to launch scripts at pre-determined times.

List $129.00 Our Price **$119.99**

PreFab Player by PreFab Software, Inc.

- Lets your scripts query and control otherwise non-scriptable applications, desk accessories and control panels.
- Adds verbs that directly manipulate the Macintosh user interface: choose from menus & pop-ups, select radio buttons, type text, determine the name of the frontmost window, the state of a check box, etc. Balloon help identifies non-standard dialog items.
- Offers a simple, inexpensive way to distribute scripts and updates.
- Use a standard scripting system for automated testing.
- Requires: AppleScript or Frontier

Our Price **$95.00**

ScriptWizard™ 1.5

• Combines the power of a professional development environment with the Mac's ease of use. • Compatible with all Apple® Open Scripting Architecture languages, including AppleScript™. • Includes delivering testing and debugging facilities to improve your productivity. • Emphasizes features that speed script development. • Single-step scripts with true statement-level stepping, watch variable values as scripts execute, jump instantly to frequently used places in a script and find and replace specific text.

List $89.00 Our Price **$84.95**

Scripter®

- Powerful and easy to use, for both novices and experts.
- Includes Superior vocabulary access – point-and-click assembly of commands and object specifications; command window for experimentation.
- Includes Shortcuts and extended editing capabilities – extensive drag-and-drop, six-function find-and-replace; navigation markers; script library collection facility; many other timesavers for faster scripting.
- Interactive debugging – comprehensive variable watcher, expression evaluation, enhanced trace log, and real single step debugging!

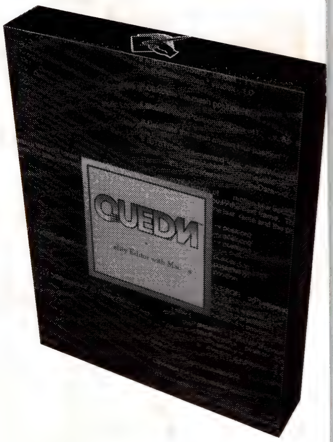
List $199.00 Our Price **$179.00**

SEE RELATED CATEGORY:
Internet Related

BBEdit 4.0 from Bare Bones Software

- A powerful, easy-to-learn text editor.
- Adds new features for HTML coders, including a spelling checker and HTML tag palette
- Accelerated for Power Macintosh; dragging supported everywhere; Internet Config aware; PowerTalk aware.
- Integrated support for Symantec's IDE, Metrowerks CodeWarrior, THINK Reference 2.x, MPW ToolServer, and most other environments.
- Many UNIX style tools, including "grep" searches, file comparisons, and sorting. Multi-file search and replace.
- PopUpFuncs feature lets you jump to a function from a menu.

List $119.00 Our price **$94.00**



QUED/M 3.0 by Nisus Software

- The programmer's text editor that defined the industry standard for speed and efficiency.
- Power PC native.
- Features integrated support for Symantec C/C++, Metrowerks CodeWarrior 6, and MPW.
- Supports all the major development environments on the Macintosh.
- Dozens of powerful editing features, including unlimited undo and redo, macro language, scripting, text folding, ten editable/appendable clipboards, markers, displaying text as ASCII codes, dynamic coloring of C/C++ keywords/comments, rectangular and non-contiguous selection.
- Includes Celestin Company's APPRENTICE 4.

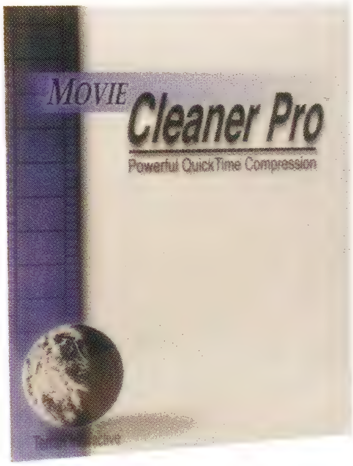
List $149.00 Our Price **$89.00**

CMaster 2.0 by Jersey Scientific

- Installs into THINK C 5 / 6 / 7 and Symantec C++ for Macintosh and enhances the editor.
- Use its function popup to select a function and CMaster takes you right to it!
- Multiple clipboards and markers, a Function Prototyper, and a GoBack Menu which can take you back to previous editing contexts.
- Almost all features bindable to the keyboard, along over a hundred keyboard-only features like "Add New Automatic Variable." Glossaries, AppleScript and ToolServer support, Macros, and External Tools you create too!

Our Price **$129.95**

SEE RELATED PRODUCTS:
Quick Time Starter Kit



Movie Cleaner Pro 1.2 by Terran Interactive

- Compress QuickTime movies
- Powerful and easy-to-use
- Includes: drag and drop batch processing, suspend and resume, high quality crop and resize, A/V fades, and much more.
- Automatically suggests the best settings for your application.
- Great for both beginners and experts
- Essential for CD-ROMs or Web sites

Our Price **$189.95**

SEE RELATED PRODUCTS:
Quick Time Official Guide

One & 2-Day SHIPPING
(call for details)

CodeManager by Metrowerks

SEE RELATED CATEGORY:
Dev. Environments

- Source code control system for the Macintosh.
- Based on and compatible with Microsoft Visual SourceSafe version 4.0.
- Works with any type of binary file including graphic, database, library and executable files.
- Includes support for multi-platform projects, security features, reverse delta versioning of files, comment areas for each revision, project analysis and reporting tools.
- Sold on a subscription basis with two future product updates.
- **Includes a 1 year MacTech subscription.**

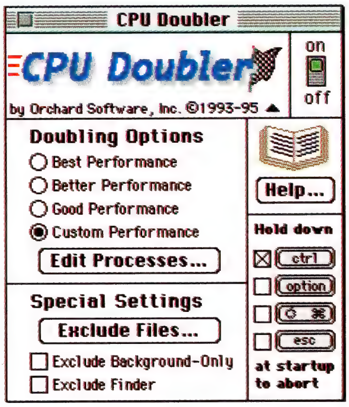
Our Price **$399.00**

CPU Doubler by Orchard Software

SEE RELATED PRODUCTS:
Cron Manager

- Performance enhancement utility for the Macintosh.
- Increases the speed of your computer by 100%.
- Works on both the PowerPC and 68K Macintosh.
- Manages computer throughput using a proprietary scheduling algorithm.
- Ensure optimal performance and compatibility.

Our Price **$79.95**



CPU Doubler

CPU Doubler
by Orchard Software, Inc. ©1993-95

on
off

Doubling Options
○ Best Performance
○ Better Performance
○ Good Performance
● Custom Performance
Edit Processes...

Help...

Hold down
☒ ctrl
☐ option
☐ ⌘
☐ esc
at startup to abort

Special Settings
Exclude Files...
☐ Exclude Background-Only
☐ Exclude Finder

CLImate by Orchard Software

- Command line interface that lets you communicate with your Macintosh using English commands to create, delete, rename, and move files and folders.
- Start applications, format disks, restart your computer, etc.
- Supplements the Finder.
- Includes a BASIC interpreter that lets you script your Macintosh without AppleScript.
- Includes advanced programming constructs: repeat loops, if/then/else conditionals, subroutine calls, etc.
- Implements wildcard characters, enabling you to work on groups of files.
- Comes bundled with sample programs and full documentation.
- Includes a free copy of Cron Manager – chronological event management utility.

Our price **$59.95**

File Genie Pro by Duet Development

FILE genie PRO

- File Genie Pro lets you search ALL your developer CD-ROMs at once!
- Every Dev CD, OS SDK, CodeWarrior CD, E.T.O., and Bookmark CD you insert is cataloged automatically. A typical Dev CD with 10,000 files is cataloged in under 10 seconds.
- You can also catalog opticals, SyQuests, and floppies.
- File Genie Pro quickly finds any file on any disk by searching hard disks, network servers, and catalogs. Open, show, print, and more. View text and graphics files without opening other applications.
- When acting on an ejected disk file, File Genie Pro tells you which disk to insert, then continues automatically.

Our Price **$89.00**

Order Toll-free
800-MACDEV-1
(800-622-3381)

Guide Composer™ 1.2 by StepUp Software

NEW!

SEE RELATED PRODUCTS:
Real World AppleGuide

SEE RELATED PRODUC
AppleGuide Complete

SEE RELATED PRODUC
Danny Goodman's AppleGuide Starter Kit

- Create powerful Apple Guide help systems for any new or existing Macintosh application.
- Great for commercial developers, shareware developers, in-house developers, and consultants.
- Provides a WYSIWYG development environment: Guide content is developed in Guide windows.
- Design topics, phrases, and panels in the same format as the user will use them. Features are WYSIWYG interface, Topics, phrases, and hierarchical phrases, Coach marks, Fully-Integrated with Apple's Guide Maker (distributed with Guide Composer), compiles scripts automatically, PICTs in Panels, Generated Guide scripts are modifiable.
- Compiled files are 100% AppleGuide-compatible and royalty-free. Easy-to-use.

New features include:

- Add up to six radio buttons on an Apple Guide panel, each of which can branch to another sequence or topic
- Add up to six checkboxes on an Apple Guide panel, each of which can insert another sequence or topic into the current topic and much more!
- FREE Update to all registered Guide Composer users. Demo is available at http://www.guideworks.com/.

Our Price **$99.00**

ScriptGen Pro by StepUp Software

- Installer script generator which requires no programming or knowledge of Rez.
- Supports StepUp's InstallerPack, StuffIt compression, custom packages, splash screens, network installs, Rez code output, importing resources, and AppleEvent link w/MPW

Our price **$169.00**

Step-Up Installer Pack by StepUp Software

- Package of several Installer "atoms" that let developers incorporate graphics, sounds, file compression and custom folder icons into installation scripts.
- Compression formats supported are Compact Pro & Diamond.
- Each atom also available separately.
- Compression requires additional licensing.

Our price **$219.00**

CronManager
by Orchard Software

SEE RELATED PRODUCTS:
CPU Doubler

- Implements the UNIX Cron facility.
- Open any Macintosh file on a given date and time.
- Simple interface.
- Works with any Macintosh file.
- Cron Manager bundled with CLImate,

Our price **$26.95**

Tenon Ported Application CD

- Contains hundreds of applications that have been pre-compiled to run on Personal or Professional MachTen 68k systems, such as Unix, Perl, and TCL/TK
- Includes many internet servers such as Gopher, WAIS Ported Application CD Vol. 1 (for 68K Macs).

Our price **$49.95**

Rosanne

SEE RELATED CATEGORY:
Scripting

- A collection of utilities which offer the user complete control over raw data.
- Sort files, extract selected records, summarize frequency counts, create sample files, perform matching on multiple files, and reformat data to new specifications, all on the desktop, and even on files of a million records or more.
- Supports AppleScript™, enabling the user to link several actions together to complete an entire process.
- Recordable – users may perform a series of actions, and see them translated directly into AppleScript commands.
- Supports multi-tasking and background processing.
- Rosanne Utilities: Copy, Format, Select, Recode, Sort, Match, and Aggregate

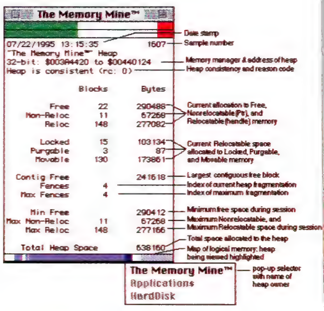
Our Price **$595.00**

Memory Mine by Adianta

SEE RELATED CATEGORY:
Dev. Environments

- Monitor heaps, identify problems such as memory leaks, and stress test applications
- Active status of memory in a heap is sampled on the fly: allocation in non-relocatable (Ptr), relocatable (Handle) and free space is shown, as are heap corruption, fragmentation, and more
- Allocate, Purge, Compact, and Zap memory lets users stress test all or part of a program
- Works on Macintoshes with 68020 or later and System 7.0 or later.

List $99.00 Our Price **$94.99**


The Memory Mine™
Date stamp
Sample number
Memory manager & address of heap
Heap consistency and reason code
Current allocation to Free, Nonrelocatable (Ptr), and Relocatable (handle) memory
Current Relocatable space allocated to Locked, Purgable, and Movable memory
Largest contiguous free block
Index of current heap fragmentation
Index of maximum fragmentation
Minimum free space during session
Maximum Nonrelocatable, and
Maximum Relocatable space during session
Total space allocated to the heap
Map of logical memory: heap being viewed highlighted
pop-up selector with name of heap owner
The Memory Mine™
Applications
HardDisk

```
07/22/1995 13:15:35          1607
"The Memory Mine™" Heap
32-bit: $003A4420 to $00440124
Heap is consistent (rc: 0)

              Blocks       Bytes

       Free       22      290488
  Non-Reloc       11       67268
      Reloc      148      277082

     Locked       15      103134
   Purgable        3          87
    Movable      130      173861

Contig Free               241618
     Fences        4
 Max Fences        4

   Min Free               290412
Max Non-Reloc     11       67268
  Max Reloc      148      277166

  Total Heap Space        638160
```

QC by Onyx Technology. High performance runtime stress testing for applications.

- Tests include heap checks, purges, scrambles, handle/pointer validation, dispose/release checks, write to zero, de-reference zero as well as other tests like free memory invalidation and block bounds checking.
- Extremely user friendly – ideal for non-programmer testers.
- Also available in Japanese.

List $99.00 Our price **$94.99**

SEE RELATED CATEGORY: **Dev. Environments**

Last Resort Programmers Edition

- Records every keystroke, command key and mouse event (in local coordinates) to a file on your hard disk
- Great for program testing & debugging, and for technical support and help desks
- Last Resort keystroke files and recovers what you typed – great for data losses caused by power failure, crashes, or accidental deleting.

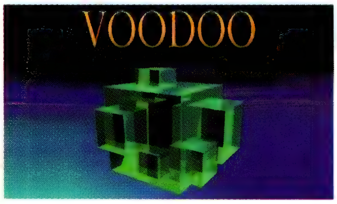
Our price **$74.95**

Spyer by InCider

- Easy to use tool that records all actions (including mouse movement) you perform on a Macintosh computer and then replays them at your preferred speed.
- Recorded data can be saved in files for future use.
- Works as a background process with any Macintosh application and is triggered by user defined Hot Keys.
- Enables the "Continuous Redo" utility and is especially useful for software testing and demonstration.

Our price **$39.00**

VOODOO



- Version control tool for the simple and clear management of projects in which files are created in numerous versions (variants and revisions).
- Allows both variant and revision control, and it manages not only variants and revisions of single files, but of a whole software project (multi files, multi users, multi variants, access rights, etc.)
- Graphical user interface and is not only suitable for mere source code control but can handle all different kinds of files with amazing compression rates: typical size of delta between arbitrary files 5%
- Please note special prices for multiple copies:

Single license **$229.00**; 2 pack **$359.00**; 5 pack **$799.00**; 10 pack **$1369.00**; 20 pack **$2399.00**

Additional pricing available on request.

SoftPolish CD-ROM by Bare Bones Software

- The essential tool for software quality assurance on the Macintosh
- Helps you identify inconsistencies with Apple's user interface guidelines, misspelled words, missing resources, and other mistakes

- Provides tools to put the finishing touches on software distribution packages prior to release
- Works independently of any programming language or environment
- Ideal for sanity checking software throughout the development process.

List price: $99 Our price: **$89**

LJ Profiler by Lars Jordebo Datakonsult

- Supports profiling of C++ 68K and PowerPC applications compiled with CodeWarrior, CFront or Symantec C++.
- Based on active profiling, i.e. profiling code called at function enter and exit, the browser application lets you follow call chain timings in hierarchical views or separate windows.
- Collect, organize, compare and save profiling data from different versions of your application into a project. Scriptable and recordable with full access to most internal data structures.
- Optional remote profiling and tracking of segment and stack usage. Full source code to what you link into your application.

Our price **$295.00**

PowerTap™

- Accelerates software by tapping into multiple processors.
- Easy API – submit tasks and retrieve results!
- Full error recovery system plus scheduling algorithms for optimal assignments and fastest possible execution.
- Compatible with all Macintosh hardware, software and major compilers.

PowerTap/2, 2 remotes edition	List $1,200.00	Our Price **$1,099.99**
PowerTap/5, 5 remotes edition	List $1,900.00	Our Price **$1,799.99**
PowerTap/n, unlimited edition	List $2,700.00	Our Price **$2,499.99**

Check out hundreds of more products on our Web site: http://www.devdepot.com

StoneTable

- Replaces all functions found in list manager
- Variable size columns/rows; different font, size, style, forecolor, backcolor per cell; sort, resize, move, copy, hide columns/rows; edit cells/titles in place; titles for columns/rows; multiple lines per cell; grid line pattern/color; greater than 32k data per table; up to 32k text per cell; support for balloon help and binary cell data. Versions for Think C, Think Pascal, MPW C, MPW Pascal, CodeWarrior 6 C.

StoneTable Extra

- Drag selected cells within table or to other tables.
- Add rows as part of drag; popup menus or check boxes in cells; variable width grid lines; move/drag/resize table in window; clipboard operations on multiple cells.
- Requires StoneTable.

This product can now be ordered in 3 different packages.

68K StoneTable

This includes StoneTable, StoneTableExtra for Think Pascal, Think C, MPW C, MPW Pascal, CodeWarrior C, CodeWarrior Pascal

Our price **$175**

PPC StoneTable

This includes StoneTable PPC, StoneTableExtra PPC for Think C, MPW C, MPW Pascal, CodeWarrior C, CodeWarrior Pascal

Our price **$175**

68K/PPC StoneTable

Includes both packages

Our price **$325**

PictureCDEF 1.3 by Paradigm Software

- Professional-level CDEF for creating custom graphical buttons (8-64 pixels) – used in products by Adobe, ProVue, STF Technologies and others!
- Multi-monitor and bit-depth sensitive.
- The button graphic (cicn, ResEdit) can be changed at runtime and even animated with a call-back routine.
- Create distinct buttons in seven variations: MultiState, PushButton, FlexiButton, ToggleButton, ChkButton, PushPictButton and TogglePictButton.
- Manual, sample code and MacApp 3.0 support included.

Full source code: **$95.00** Object code: **$45.00**

SpellsWell 7 1.0.4

- Award-winning, comprehensive, practical spelling checker that works in batch mode or within applications that incorporate the Apple Events Word Services protocol (e.g., Eudora, WordPerfect, Communicate!, and Fair Witness).
- Checks for spelling errors as well as common typos like capitalization errors, spaces before punctuation, double word errors, abbreviation errors, a/an before vowel/consonant, etc.
- MacTech orders include developer kit with Writeswell Jr., a sample AppleEvents Word Services word-processor and its source code.

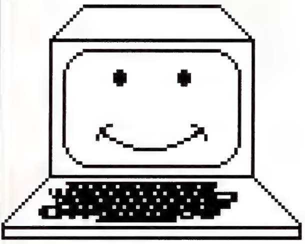
Our price **$74.95**

MacWireFrame by Amplified Intelligence

- Create your own virtual reality application with MacWireFrame, a virtual reality application framework.
- Includes a complete library of object oriented graphics routines.
- Easy to understand application frame work, plus an example application program that lets you start solid modeling right away.
- Complete with fully documented source code.
- Guaranteed $49.99 upgrade to the soon to be released, scriptable, MacWireFrame 5.0.

List $299.00 Our Price **$75.00**

Order Toll-free
800-MACDEV-1
(800-622-3381)



B-Tree HELPER™ 2.2

- Inexpensive database engine for Macintosh programmers in C source code.
- Uses contiguous fixed length blocks.
- Expands the file as necessary and contracts files when possible.
- Inserts and deletes keys in one or more B-Trees.
- Finds keys equal to, less than, or greater than a given value in a few hundredths of a second.
- Finds lists of records whose keys are equal to, less than, or greater than a given value or are in a range of values.

Our Price **$150.00**

Q3S/3dPane/SmartPane

ViviStar CONSULTING

- Full featured 3d graphics library.
- Polyhedra; Gouraud shading; stereoscopic projections; pipeline access; animation and model interaction support; a “triad mouse” to map 2d mouse movement to 3d; and all the regular 3D features.
- 3dPane provides integration with the TCL and provides a view orientation controller. SmartPane provides TCL offscreen image buffering, flicker free animation, and QuickTime movie recording. SmartPane functions in 3d or 2d scenarios.
- All work with C++ compilers or ThinkC 6 and compile to PowerPC or 68K target machines.

Our Price **$192.00**

dtF

- True relational database system for Apple Macintosh computers.
- Provides a powerful choice for developers who want to create database centered applications with no performance trade-offs.
- Features SQL, full transaction control, error recovery, single user, client server architecture and multi-platform support including DOS, Windows, OS/2 and UNIX.
- The C/C++ API is identical and fully portable across all supported platforms.
- Third-party vendors supporting dtF will be able to offer a variety of advanced features and benefits to their customers royalty free.
- Tools are included for importing, exporting, creating and managing databases and users.
- Supported development environments include: Symantec, MPW, Metrowerks and more. Mac/SDK

List $695.00 Our Price **$679.99**

OOFILE

SEE RELATED PRODUCTS:

OOFILE HTML Writer

- First OODBMS framework to offer a complete solution for application authors.
- Replaceable backend database, currently Faircom’s c-tree Plus for cross-platform royalty-free power.
- PowerPlant and other frameworks integrated with edit fields, database browsers & more. AppMaker users, generate complete applications.
- User-friendly syntax makes it easier to migrate from FoxPro and the non-OO world.
- **$900.00** for a once-off c-tree bundle, or **$1,095.00** 1-year subscription.
- HTML and character-mode report-writer. Our Price **$195.00**.
- Full GUI report-writer, including HTML. Our Price **$495.00**.
- Mac Platform Bundle - includes all Mac frameworks, c-tree, and Report-Writer. Our Price **$1,495.00**

NeoAccess™

- Full-featured object database engine for use in Macintosh, Windows, Unix and DOS based C++ applications.
- Extended binary trees and binary search algorithms tuned for short access times; dynamically combined, collapsed, and compressed indices; object caching for instant access to previously used objects.
- Build fast, powerful applications in record time!

Our price **$749.00**

Lowest price guaranteed.

Feel Good about your purchase!

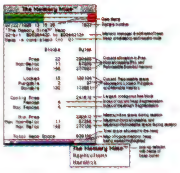
30-day return policy.

AppMaker

- Develop the user interface for a Macintosh application using the original interface builder.
- Just point and click to design your application.
- Creates resources and generates excellent source code.
- Supports most development environments including Metrowerks, Symantec, or MPW; C, C++, or Pascal; procedural or object-oriented, using PowerPlant, TCL, or MacApp.
- The generated code uses the Universal Headers to provide PowerMac compatibility.
- Great tool for beginners to learn object-oriented and Macintosh Toolbox programming techniques.
- Includes one-year subscription on CD.

Includes CD-ROM!

List $299.00 Our Price ~~$279.99~~ SPECIAL PRICE! **$149.00**



MacDesigner/Expert by Excel Software

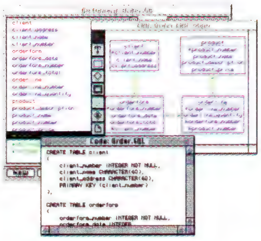
- Supports software engineering methods with the capabilities of MacDesigner plus multi-task design.
- Integrated requirement database provides traceability from requirement statements to design diagrams, code or test procedures.
- Suited to design or maintenance of real-time, multi-tasking software projects.

Expert Demo **$79.00**; Expert Product **$1595.00**; MacDesigner Demo **$79.00**; MacDesigner **$995.00**

MacAnalyst/Expert by Excel Software

- Supports software engineering methods with the capabilities of MacAnalyst plus state transition diagrams, state transition tables, decision tables and process activation tables
- An integrated requirement database provides traceability from requirement statements to analysis or design diagrams, code or test procedures
- Well suited to the analysis and design of real-time or requirements driven projects.

Expert Demo **$79.00**; Expert MacAnalyst **$1595.00**; MacAnalyst Demo **$79.00**; MacAnalyst **$995.00**



```
CREATE TABLE client
(
    client_number INTEGER NOT NULL,
    client_name CHARACTER(40),
    client_address CHARACTER(40),
    PRIMARY KEY (client_number)
);

CREATE TABLE orderform
(
    orderform_number INTEGER NOT NULL,
    orderform_date INTEGER
```

MacA&D by Excel Software

- Combines the capabilities of MacAnalyst/Expert and MacDesigner/Expert into a single application.
- Supports structured analysis and design, object-oriented analysis and design, real-time extensions, task design, data modeling, screen prototyping, code editing and browsing, reengineering, requirement traceability, and a global data dictionary.

MacA&D Demo **$149.00**; MacA&D **$2995.00**

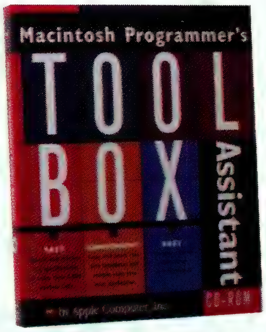
ICONIX Power Tools - Ten modules that can be used either as stand-alones or combined to form a custom, integrated development environment with dictionaries and Language Sensitive Editors

- **ObjectModeler** supports several object-oriented methods with four graphical editors
- **FastTask** provides real-time extensions in the form of State Transition Diagrams and their equivalent matrix representation
- **DataModeler** supports object-oriented analysis and includes an Entity Relationship Attribute (ERA) Diagram Editor
- **FreeFlow** provides support for DeMarco Structured Analysis with real-time extensions and several object-oriented methods.
- **PowerPDL** supports detailed algorithm design using pseudocode.
- **AdaFlow** supports Ada-oriented design and Ada development
- **QuickChart** supports Structure Chart Editing
- **SmartChart** supports Detailed Design with automatic Structure Chart Generation from Program Design Language, Processes Hierarchy Charts with FreeFlow
- **ASCII Bridge** enables PowerTools repository information to be merged, imported, and exported in ASCII or CDIF format and provides an Interleaf interface.
- **CoCoPro** implements Boehm's Constructive Cost Model (CoCoMo) technique for estimating costs of software projects.

PowerTools is available both as individual modules or in PowerPack bundles of 6, 8 or 10 distinct modules of your choice: Each individual module is **$1,495.00**.
PowerPack/6: **$5,995**; PowerPack/8: **$6,995.00**.
PowerPack/10: **$7,995.00** (Full Product Line).
Upgrade Service: 15% of purchase price, good for one calendar year upgrade.
Training and Consulting: **$3,000.00** per day, plus expenses.

Inside Macintosh®: Programmer's Toolbox Assistant CD-ROM

Instant electronic access to Inside Macintosh essentials.



- Get quick access to reference pages for over 4,000 Toolbox calls in your system software from their development environment.
- Essential information for Macintosh software developers.
- Hypertext links allow programmers to view related topics easily.
- The ultimate electronic reference tool for Macintosh programmers.

List $99.95 Our Price **$89.95**

Inside Macintosh®: Overview by Apple Computer, Inc.

- An overview of Macintosh programming fundamentals and a road map to the New Inside Macintosh library.
- Covers various programming tools and languages, compatibility guidelines and considerations for worldwide development. 176 pages.
- Great for beginners!

List $22.95 Our Price **$20.65**

Inside Macintosh®: Macintosh Toolbox Essentials

by Apple Computer, Inc.

- Covers the heart of the Macintosh. The toolbox enables programmers to create applications consistent with the Macintosh "look and feel".
- Describes Toolbox routines and shows how to implement essential user interface elements, such as menus, windows, scroll bars, icons and dialog boxes. 880 pages.

List $34.95 Our Price **$31.45**

Inside Macintosh®: More Macintosh Toolbox by Apple Computer, Inc.

- Managers discussed include Help, List, Resource, Scrap and Sound.
- Covers other Macintosh features such as how to support copy and paste, provide Balloon Help, play and record sound and create control panels.

List $34.65 Our Price **$31.45**

Inside Macintosh®: Files by Apple Computer, Inc.

- Describes the parts of the operating system that allow you to manage files.
- Explains how your application can handle the commands typically found in a File menu.
- Provides a reference to the File and Alias Managers, the Disk Initialization and Standard File Packages. 510 pgs.

List $29.95 Our Price **$26.95**

Inside Macintosh®: Operating System Utilities by Apple Computer, Inc.

- Describes parts of the Macintosh Operating System that allow you to manage various low-level aspects of the operating system.
- Shows how to get information about the operating system, manage operating system queues, handle dates and times, control the settings of the parameter RAM, manipulate the trap dispatch table, and receive and respond to low-level system errors.

List $26.05 Our Price **$23.45**

Inside Macintosh®: Processes by Apple Computer, Inc.

- Describes the parts of the Macintosh operating system that allow you to control the execution of processes and interrupt tasks.
- Shows how you can use the Process Manager to get information about processes loaded in memory.
- A reference for the Vertical Retrace, Time, Notification, Deferred Task, and Shutdown Managers. 208 pages.

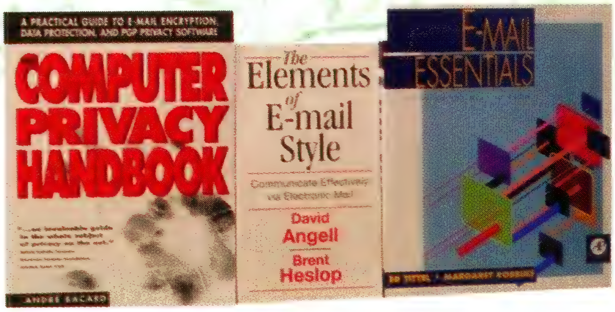
List $22.95 Our Price **$20.65**

Order Toll-free
800-MACDEV-1
(800-622-3381)

Inside Macintosh®: Memory by Apple Computer, Inc.

- Describes the parts of the Macintosh operating system that allow you to manage memory.
- Provides detailed strategies for allocating and releasing memory, avoiding low-memory situations, reference to the Memory Manager, the Virtual Memory Manager, and memory-related utilities. 296 pages.

List $24.95 Our Price **$22.45**



E-Mail Essentials by Ed Tittel & Margaret Robbins

- A hands-on guide to the basics of e-mail.
- Covers everything from the installation of e-mail to the maintenance and management of e-mail hubs and message servers.
- For the casual e-mailer and the networking professional. 250 pp.

List $24.95 Our Price **$22.45**

The Computer Privacy Handbook

- A practical guide to e-mail encryption, data protection, and PGP privacy software.
- Learn the practical steps you can take to safeguard your electronic security.

List $24.95 Our Price **$22.45**

The Elements of E-Mail Style

by Brent Heslop and David Angell

- The rules of the road in the e-mail age.
- How to write solid, effective E-Mail and avoid common pitfalls. 208 pages.

List $14.95 Our Price **$13.45**

Lowest price guaranteed. Feel Good about your purchase! 30-day return policy.

Cyberpunk Handbook, The Real Cyberpunk Fakebook

by St. Jude, R.U. Sirius, and Bart Nagel. Published by Random House.

Are you or someone you know a Cyberpunk? Read this bookand you'll know!

List $9.95 Our Price **$8.95**

Java in a Nutshell

- A complete quick reference guide to Java, the hot new programming language from SunMicrosystems.
- Contains descriptions of all of the classes in the Java 1.0 API, with a definitive listing of all methods and variables.
- Also contains an accelerated introduction to Java for C and C++ programmers who want to learn the language fast.

List $14.95 Our price **$13.45**

Planning and Managing Websites

Includes CD-ROM!

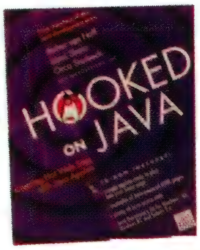
- The definitive guide to setting up and running a Web site on the Macintosh.
- Learn everything you need to know about using WebSTAR, the best known HTTP server software and its shareware predecessor MacHTTP.
- Write CGI applications for your server – in AppleScript and in C.
- CD includes a special version of WebSTAR, plus tons of useful software.

List $39.95 Our Price **35.96**

SEE RELATED CATEGORY: **Internet Related**

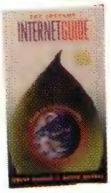
Hooked on Java

- Written by the Java development team at Sun.
- An introduction to using applets, for Web administrators, designers, and developers.



- Demonstrates how to use applets in your own pages.
- Includes a concise introduction to the Java language, and a CD with tools.

List $29.95 Our Price **$26.95**



The Instant Internet Guide

- An Internet jump-start – how to access, use and navigate global networks, 224 pages.

List $14.95 **$13.45**

Teach Yourself Java for Macintosh in 21 Days

Includes CD-ROM!

- Add interactivity and multimedia to Web pages!
- Astep-by-step guide to make your Web site come alive.
- Learn the basics of programming Java applets and the concepts behind the Java language.

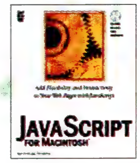
- Includes CD-ROM with a limited version of Roaster, the first commercial, integrated applet development environment for Java for the Macintosh!

List $40.00 Our price **$36.00**

JavaScript for the Macintosh

Includes CD-ROM!

- Allows non-programmers to take advantage of the power of Netscape Navigator.
- Expand the capabilities of your Web page, without having to understand C or C++.
- CD-ROM contains "Wizlets" that allows you to easily create your own JavaScripts



- Takes you step-by-step through programming cross-platform JavaScripts
- Details how to create JavaScripts for JavaScript-aware Web browsers

List $45.00 Our price **$40.50**

Developer DEPOT™
ORDER FORM

http://www.devdepot.com
E-mail: orders@devdepot.com
U.S./Canada: 800-MACDEV-1 (800-622-3381)
Outside U.S./Canada 805-494-9797
Fax: 805-494-9798

__ 3D Game Machine	$299.00
__ 3D Graphics w/QD 3D	$26.95
__ Adv. Color Imaging/Mac	$33.25
__ AppleGuide Complete	$35.96
__ AppleScript Finder Guide	$17.95
__ AppleScript Lang. Guide	$26.95
__ AppleScript Scripting Add.	$17.05
__ Applied Mac Scripting	$31.45
__ AppMaker	$149.00
__ B-Tree Helper	$150.00
__ BASIC for the Newton	$99.00
__ BBEdit	$94.00
__ Black Art of Game Prog.	$35.99
__ C++ Prog. w/MacApp	$31.46
__ C++ Prog. with CW	$32.35
__ CLImate	$59.95
__ CMaster	$129.95
__ CodeManager	$399.00
__ CodeWarrior Bronze	$149.00
__ CodeWarrior Gold	$399.00
__ CW Mouse Pads (Arnold)	$8.95
__ CW Mouse Pads (Gears)	$9.95
__ CW Wear (Arnold at work)	$9.95
__ CW Wear (Black Sweat)	$29.95
__ CW Wear (Blood, Sweat…)	$7.95
__ CW Wear (Cross Platform)	$14.95
__ CW Wear (Hat)	$14.95
__ CW Wear (Hawaii Five-O)	$7.95
__ CW Wear (Winter Hat)	$14.95
__ Complete AppleScript	$31.50
__ Complete HyperCard	$31.50
__ Computer Privacy	$22.45
__ CPU Doubler	$79.95
__ CronManager	$26.95
__ Cyberpunk Handbook, …	$8.95
__ Dan Shafer … Prograph	$44.95
__ Danny Goodman's AppleGuide Starter Kit	$31.46
__ Danny Goodman's AppleScript Handbook	$35.00
__ DataScript	$229.00
__ Discover Prog. for Mac	$79.00
__ dtF	$679.99
__ E-Mail Essentials	$22.45
__ Elements of E-Mail Style	$13.45
__ Essential OpenDoc	$35.95
__ FaceSpan	$179.00
__ File Genie Pro	$89.00
__ Fortran 77SDK	$699.00
__ Fragment of Imagination	$35.96
__ FrameWorks Magazine	$8.00
__ FrameWorks Sources Disk	$10.00
__ Graphic Gems V	$44.95
__ Guide Composer	$99.00
__ Hooked on Java	$26.95
__ HyperCard Stack Design	$19.95
__ HyperTalk 2.2: The Book	$31.50
__ ICONIX Power Tools	Call
__ Inside CodeWarrior 8	$34.95
__ IM: AOCE Appl. Interfaces	$24.25
__ IM: AOCE Srvce Acc Mod	$36.40
__ IM: CD-ROM	$89.95
__ IM: Devices	$26.95
__ IM: Files	$26.95
__ IM: Imaging	$24.25
__ IM: Interapp Comm.	$31.45
__ IM: Mac Toolbox Essntials	$31.45
__ IM: Memory	$22.45
__ IM: More Mac Toolbox	$31.45
__ IM: Networking	$26.95
__ IM: Operating Sys Utils	$23.45
__ IM: Overview	$20.65
__ IM: PowerPC Numerics	$26.00
__ IM: PowerPC Sys Software	$22.45
__ IM: Processes	$20.65
__ IM: QD GX Env/Utilities	$26.95
__ IM: QD GX Graphics	$24.25
__ IM: QD GX Objects	$24.25
__ IM: QD GX Printing	$24.25
__ IM: QD GX Printing Ext	$26.95
__ IM: QD GX Prog. Overvw	$22.45
__ IM: QD GX Typography	$26.95
__ IM: QuickTime	$26.95
__ IM: QuickTime Compnts	$31.45
__ IM: Sound	$24.25
__ IM: Text	$35.95
__ IM: X-Ref	$17.95
__ Inside PowerPlant Manual	$34.95
__ Inside Mac Comm Toolbx	$22.95
__ Instant Internet Guide	$13.45
__ Java in a Nutshell	$13.45
__ JavaScript for the Mac	$40.50
__ Last Resort Prog. Edition	$74.95
__ Learn C on the Mac	$31.45
__ Learn C++ on the Mac	$33.26
__ LJ Profiler	$295.00
__ LPA MacProlog	Call
__ LS Fortran	$695.00
__ LS Object Pascal CD	$349.95
__ Mac Bathroom Reader	$11.70
__ MacDesigner, MacAnalyst, or MacA&D	Call
__ MacFortran II	$549.99
__ MachTen Power Unix	$695.00
__ MachTen XWindows	$350.00
__ Mac C Prog. Primer Vol. 1	$24.25
__ Mac C Prog. Primer Vol. 2	$24.25
__ Mac OLE2 Prog. Ref.	$26.95
__ Mac Pascal Prog. Primer-I	$24.25
__ Mac Programming Secrets	$28.76
__ Mac Programming Tech	$31.95
__ MacsBug Ref & Debug.	$31.46
__ MacTech CD-ROM	$49.00
__ MacTech CD (Upgrade)	$39.00
__ MacTech Magazine (Can)	$59.00
__ MacTech Magazine (Int'l)	$97.00
__ MacTech Magazine (US)	$47.00
__ MacTech Mag Back Issues	$5.00
__ MacTech Mouse Pad	$8.95
__ MacTutor, Best of	$9.95
__ MacWireFrame	$75.00
__ MADACON '93 CD-ROM	$9.95
__ Mastering THINK Class Lib.	$26.95
__ Memory Mine	$94.99
__ Metrowerks CW Prog	$35.95
__ Mjølner BETA System	$249.99
__ More Mac Prog. Tech.	$35.95
__ Movie Cleaner Pro	$189.95
__ NeoAccess	$749.00
__ Object Oriented Design	$20.66
__ OOFILE HTML Writer.	Call
__ OpenDoc Prog. Cookbk	$22.45
__ OpenDoc Prog. Guide	$40.46
__ Optimizing PowerPC Code	$35.96
__ PCI System Architecture	$31.46
__ Personal MacTen	$495.00
__ Picture CDEF (Obj only)	$45.00
__ Picture CDEF (Full Srcs)	$95.00
__ Planning & Managing Websites	$35.96
__ Power Mac Prog. Strtr Kit	$35.10
__ PowerPC Sys. Architecture	$31.46
__ PowerTap	Call
__ PreFab Player	$95
__ Presenting Magic Cap	$15.25
__ Profit from Experience	$22.45
__ Prog. Toolbox Asst. CD	$89.95
__ Prog. with NS BASIC	$32.35
__ Prog. with NewtonScript	$26.95
__ Prog. Symantec C++/Mac	$26.95
__ Prog. Primer Mac Vol. 1	$34.15
__ Prog. QuickDraw	$24.25
__ Prog. with AppleTalk	$26.00
__ Q3S/3dPane/SmartPane	$192.00
__ QC	$94.99
__ QUED/M	$89.00
__ QuickTime Official Guide	$45.00
__ QuickTime Strtr Kit/Mac	$40.50
__ Real World AppleGuide	$35.95
__ ResEdit All Night Diner	$22.45
__ ResEdit Complete	$31.45
__ Roaster	$299.00
__ Rosanne	$595.00
__ Script Debugger	$119.99
__ ScriptBase	$79.00
__ Scripter	$179.00
__ ScriptGen Pro	$169.00
__ ScriptWizard	$84.95
__ Sex, Lies and Video	$31.46
__ SmalltalkAgents	$695.00
__ SoftPolish CD-ROM	$89.00
__ Software by Design	$26.95
__ SpellsWell	$74.95
__ Spyer	$39.00
__ Step-Up Installer Pack	$219.00
__ 68K StoneTable	$175.00
__ PPC StoneTable	$175.00
__ 68K/PPC StoneTable	$325.00
__ Symantec C++ for 68k	$99.00
__ Symantec C++ for Power Macintosh	$349.00
__ Symantec C++ Prog.	$40.50
__ Tao of AppleScript: BMUG's Guide to Mac Scripting	$26.95
__ TCP/IP Scripting Addition	$49.00
__ Teach Yourself Java	$36.00
__ Tenon Ported App. CD	$49.95
__ THINK Pascal	$169.00
__ Tog on Software Design	$26.95
__ Tricks of the Mac Game Programming Gurus	$45.00
__ Visual Programming with Prograph CPX	$30.60
__ VOODOO (Single pack)	$229.00
__ VOODOO (2 pack)	$359.00
__ VOODOO (5 pack)	$799.00
__ VOODOO (10 pack)	$1,369.00
__ VOODOO (20 pack)	$2,399.00
__ Web Weaving	$22.45
__ WebHead Mac Guide	$22.45
__ Webmaster Macintosh	$26.95
__ Wireless for the Newton	$31.45

Ordering Instructions: Check off the products you want above. Fill in the subtotal, sales tax (for California residents only), and total. Shipping charges will be added upon order processing. All orders will be shipped via most econimical method (e.g., UPS Ground), unless you state otherwise. **Remember to fill out your shipping, and payment information on the other side.**

Subtotal ____________________ Sales Tax (8.25% for CA residents) ______________ Total ____________________

Name ______________________________

Company ______________________________

Address ______________________________

City ______________________ State/Province __________

Zip/Postal Code ____________ Country ______________________

Home Phone ______________ Business Phone ______________

E-Mail Address ______________ Fax ______________

☐ Cash ☐ Check/Money Order Check # ______________

☐ VISA ☐ MasterCard ☐ American Express Signature ______________

Card # ______________________ Expiration Date ____________

If you are ordering a subscription to MacTech Magazine, are you a new subscriber, or are you renewing your subscription?

☐ New Subscription
☐ Renewal

Please fill out as completely as possible to avoid delays in processing your order.

Fold Here (First, so panel below covers your personal information.)

Fold Here (Last) and Tape Closed (not staple)

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS MAIL PERMIT NO. 264 THOUSAND OAKS, CA

POSTAGE WILL BE PAID BY ADDRESSEE

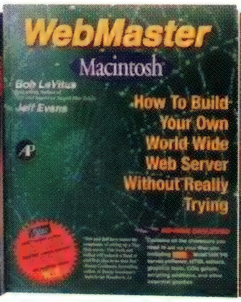
XPLAIN CORPORATION
P O BOX 5200
WESTLAKE VILLAGE CA 91359-9864

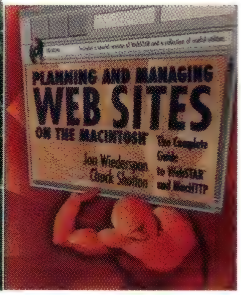
Web Weaving

- Provides the guidance to create and maintain web sites that are easily navigated, scaleable, and maintainable.
- Covers the three major platforms for Web pages – UNIX, Windows, and the Mac – shows how to use software tools and utilities that are important in setting up web sites.
- Learn how to install and configure Web servers, use authoring tools and converters, implement security schemes, and integrate multimedia features such as sound, video, and graphics into Web pages.
- Learn to organize and present the information in Web sites. Includes planning for growth, building in maintenance schemes, catering to users' needs, and creating a navigable, logical underlying infrastructure.

List $24.95 Our Price **$22.45**

SEE RELATED CATEGORY: **Internet Related**


WebMaster
Macintosh
Bob LeVitus
Jeff Evans
How To Build Your Own World Wide Web Server Without Really Trying



Webmaster Macintosh

SEE RELATED CATEGORY: **Internet Related**

- A fast and easy guide for turning your Macintosh into a World Wide Web site – in a matter of hours!
- Includes CD!

List $29.95 Our Price **$26.95**

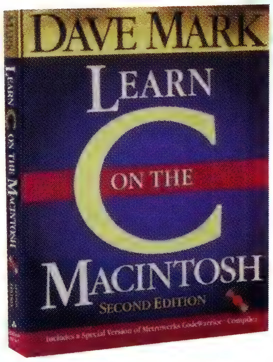
Includes CD-ROM!

WebHead Mac Guide

SEE RELATED CATEGORY: **Internet Related**

- Get the most from the Web.
- Easy-to-understand, great for beginners, and online veterans.
- Includes instructions on how to build your own home page, posting pages on the Web, and avoiding common HTML mistakes.

List $24.95 Our Price **$22.45**



Learn C on The Macintosh Second Edition By Dave Mark

- New revised edition.
- Easy-to-understand – everything you need to start programming!
- Updated and enhanced exercises that lead you step by step. You'll learn function, variables, pointers datatypes, data structures, file input and output and more!
- Includes CD-ROM with Metrowerks CodeWarrior™ Lite – the hottest Macintosh programming environment (including a PowerPC version).

List $34.95 Our Price **$31.45**

SEE RELATED CATEGORY: **Dev. Environments**

Discover Programming for Macintosh

- Includes full working version of CodeWarrior along with three online tutorial books and Dave Mark's "Learn C on the Macintosh" converted to AppleGuides.
- Includes C, C++ and Object Pascal compilers for generating 68K Macintosh code, source-level debuggers, object-oriented frameworks (PowerPlant, MacApp), Apple's MPW, complete online documentation and source code examples for all languages and platforms.
- The IDE software has been localized in eight languages plus English. This product is not sold as a subscription.
- **Includes a 3 month subscription to MacTech Magazine.**

Our Price **$79.00**

SEE RELATED CATEGORY: **Dev. Environments**

Programming Primer Macintosh Vol 1 by John Whittle and Judy May

- An introduction to Macintosh programming, using C++ as the example language
- Provides realistic, easy to follow, programming examples designed to work with either Symantec® C++ or Metrowerks® CodeWarrior™
- Includes one 3.5" disk with source code for the programming examples, plus numerous public domain utilities to use with each compiler

List $37.95 Our Price **$34.15**

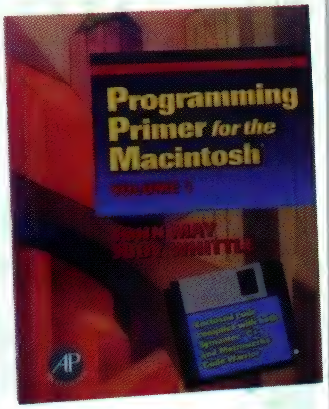
One & 2-Day SHIPPING
(call for details)

Macintosh C Programming Primer Volume I

Second Edition, Inside the Toolbox Using THINK C by Dave Mark and Cartwright Reed

- Updated new edition of the Macintosh programming best seller.
- System 7, new versions of THINK C and ResEdit.
- Learn how to use the resources, Macintosh Toolbox and interface to create stand-alone applications.
- 672 pages.

List $26.95 Our Price **$24.25**



Macintosh C Programming Primer Volume II

Mastering the Toolbox Using THINK C by Dave Mark.

- Covers advanced topics such as: Color QuickDraw, THINK Class Library, TextEdit, and the Memory Manager: 528 pgs.

List $26.95 Our Price **$24.25**

Learn C++ on the Macintosh by Dave Mark.

SEE RELATED CATEGORY: **Dev. Environments**

- Basic syntax of C++ and object programming.
- Learn how to write, edit, and compile your first C++ programs.
- Features key C++ concepts such as derived classes, operator overloading, iostream functions and more.
- Includes a special version of Symantec C++ for Macintosh. Book/disk package with 3.5" 800K Macintosh disk. 400 pages.

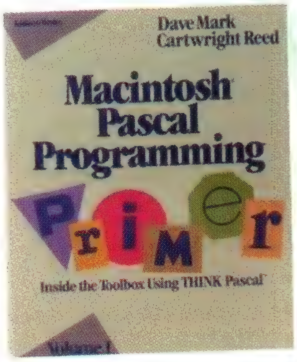
List $36.95 Our Price **$33.26**

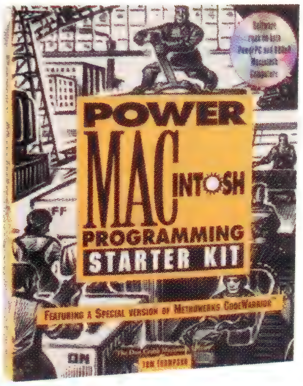
Macintosh Pascal Programming Primer Volume I

Inside the Toolbox Using THINK Pascal by Dave Mark and Cartwright Reed.

This tutorial shows programmers new to the Macintosh how to use the Toolbox, resources, and the Macintosh interface to create stand-alone applications with Symantec's THINK Pascal. 544 pages.

List $26.95 Our Price **$24.25**


Dave Mark
Cartwright Reed
Macintosh Pascal Programming Primer
Inside the Toolbox Using THINK Pascal
Volume I


POWER MACINTOSH PROGRAMMING STARTER KIT
Featuring a Special Version of Metrowerks CodeWarrior
Tom Thompson

Power Macintosh Programming Starter Kit

by Tom Thompson.

SEE RELATED CATEGORY: **Dev. Environments**

- Enter the world of the PowerPC chips.
- Get the scoop on the microprocessors, the RISC architecture, and how to write native code and emulation operations to create software for the Macintosh PowerPC.
- CD-ROM includes a unique compiler for writing code easily.

Includes CD-ROM!

List $39.95 Our Price **$35.10**

Macintosh Programming Secrets 2nd edition

By Scott Knaster, and Keith Rollin

- Macintosh Programming Secrets is divided in two parts.

Part 1, "Concepts and Ideas", discusses the evolution of the Macintosh and the standards, customs, and software that shape the system as well as the Macintosh user interface.

Part 2 "Technical Adventures", presents the skeleton of an application, and then builds upon that framework to describe how to: • Create fancy dialogue boxes • Utilize the new 32 bit QuickDraw developments • Track the mouse with "marching ants" • Manage multiple windows with the Window manager • Copy files within a program • Install the worlds strangest spinning cursor.

List $31.95 Our Price **$28.76**

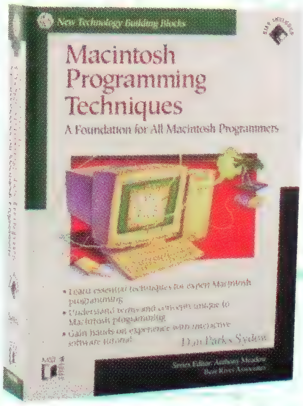
Order Toll-free
800-MACDEV-1
(800-622-3381)

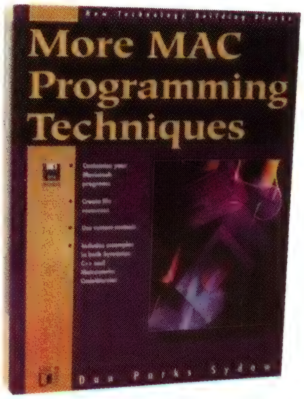
Macintosh Programming Techniques

by Dan Sydow (Series Editor: Tony Meadow)

- For Macintosh programmers as well as those making the transition from DOS, Windows, VAX or UNIX.
- Emphasizes programming techniques for better code – regardless of language!
- Guides you through Macintosh memory management, QuickDraw, events and more, using sample program in C++.
- Includes disk with an interactive tutorial, plus reusable C++ code.

List $34.95 Our Price **$31.95**


New Technology Building Blocks
Macintosh Programming Techniques
A Foundation for All Macintosh Programmers
Dan Parks Sydow


New Technology Building Blocks
More MAC Programming Techniques
Dan Parks Sydow

More Mac Programming Techniques

- Hands-on guide and tutorial, expands on the basic foundation of programming to develop truly powerful applications.
- Take a detailed look at the units of a Macintosh program – from the INITs to custom controls.
- Packed with solid tricks and techniques.
- Learn file resources from the ground up, and how to build custom menus, add custom controls, including buttons, and slider controls, MDEF, CDEF resources and much more.

List $39.95 Our Price **$35.95**

Programming with AppleTalk by Michael Pierce

- Programming with AppleTalk is the hands-on guide to understanding and working with AppleTalk.

Topics covered include:

- How to create applications and system extensions that run with AppleTalk.
- AppleTalk protocols and the protocol stack, transport media, the Preferred AppleTalk Inteface, and the storage management.
- Numerous working code examples walk you through using RDEV, INIT, NBP, ATP, and ADSP You will also learn the use of: Synchronous, and asynchronous calls, How to avoid heap fragmentation, And how to configure a Chooser Interface.

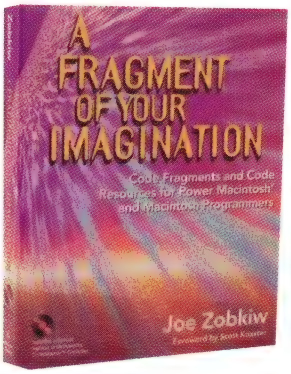
List $45.00 Our Price **$26.00**

A Fragment of Your Imagination by Joe Zobkiw

- Packed with useful code fragments for the Macintosh and Power Macintosh.
- Hard to find information about techniques used to structure and build fat, safe fat, and accelerated code resources.
- All code is reusable and is provided on the disc, along with Metrowerks Code Warrior Lite. Book/CD-ROM, 528 pages.

List $39.95 Our Price **$35.96**

Includes CD-ROM!

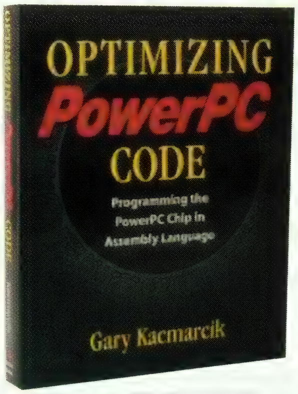

A FRAGMENT OF YOUR IMAGINATION
Code Fragments and Code Resources for Power Macintosh and Macintosh Programmers
Joe Zobkiw

Programming QuickDraw

- Learn to build color pictures on the Mac
- Learn to maintain the highest degree of compatibility for you applications across the Mac platform
- Learn to perform sophisticated image processing operations with CopyBits( ) and CopyDesk( )
- Learn to enable your applications to take advantage of QuickDraw compliant graphics hardware and accelerators.

List $26.95 Our price **$24.25**

Lowest price guaranteed. Feel Good about your purchase! 30-day return policy.



Optimizing PowerPC Code: Programming the PowerPC in Assembly Language

- Take full advantage of the potential of the PowerPC by mastering the Assembly Language techniques.
- Learn to produce faster more robust software!

List $39.95 Our Price **$35.96**

Inside Macintosh®: PowerPC Numerics

by Apple Computer, Inc.

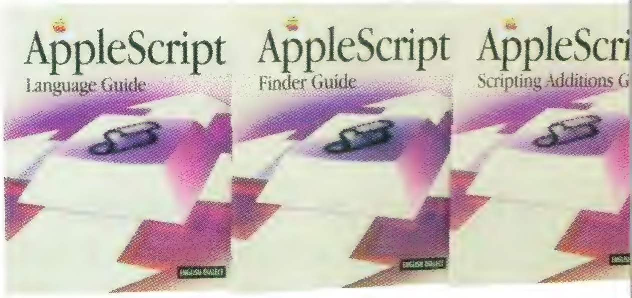
- Describes the floating-point numerics environment provided with the first release of PowerPC processor-based Macintosh computers.
- Provides a description of the IEEE standard 754 for binary floating-point arithmetic., and how RISC Numerics compiles with it.
- Shows programmers how to create floating-point values and how to perform operations on floating-point values in high-level languages such as C and in PowerPC assembly language.

List $28.95 Our Price **$26.00**

Inside Macintosh®: PowerPC System Software by Apple Computer, Inc.

- Describes the new process execution environment and system software services provided with the first version of the system software for Macintosh on PowerPC computers.
- Contains information to write applications that can run on the PowerPC.
- Shows how to make your software compatible with the new run-time environment provided on PowerPC-based Macintosh computers. It also provides a complete technical reference for the Mixed Mode Manager, the Code Fragment Manager, and the Exception Manager.

List $24.95 Our Price **$22.45**



AppleScript Finder Guide, English Dialect by Apple Computer, Inc.

- Provides definitions for Finder object classes and commands.
- Write, record, or run scripts that trigger the same desktop actions that you trigger using the keyboard and mouse.

List $19.95 Our Price **$17.95** SEE RELATED CATEGORY: **Scripting**

AppleScript Language Guide,

by Apple Computer, Inc.

- A complete reference for anyone using AppleScript to modify existing scripts or to write new ones.
- Contains useful information for programmers who are working on scriptable applications or complex scripts.
- Features detailed definitions of AppleScript terminology and syntax in the following categories: Value classes, commands, objects and references to objects, expressions, control statements, handlers, and script objects.
- Includes many sample scripts, discusses advanced topics such as writing command handlers for script applications, the scope of script variables and properties declared at different levels in a script, and inheritance and delegation among script objects.

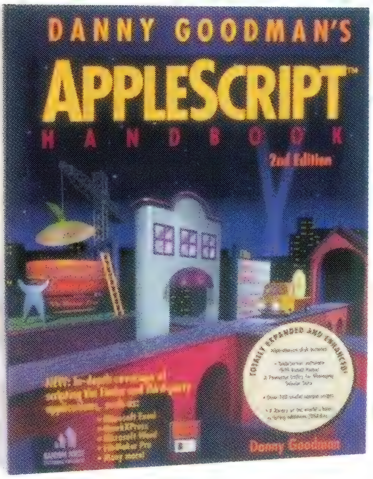
List $29.95 Our Price **$26.95** SEE RELATED CATEGORY: **Scripting**

AppleScript Scripting Additions Guide

by Apple Computer, Inc.

- Use the standard scripting additions commands.
- Write scripting additions.

List $18.95 Our Price **$17.05** SEE RELATED CATEGORY: **Scripting**



Danny Goodman's AppleScript Handbook Second Edition by Danny Goodman

- Customize and extend the capabilities of any Macintosh computer – no programming experience needed!
- Learn to use scripts to enhance the Macintosh environment, automate many processes, link data between applications, and much more.
- All-new examples showing how to integrate AppleScript with the Finder, spreadsheets, desktop publishing programs, graphics applications, databases, telecommunications programs, utilities, and HyperCard.
- Includes 3 1/2" disk with over $100 worth of software, including AppleScript 1.1, valuable utilities, and powerful, ready-to-use scripts.

Includes Disk!

List $39.00 Our Price **$35.00**

SEE RELATED CATEGORY: **Scripting** SEE RELATED CATEGORY: **Tools, Libs & Utilities**

Applied Mac Scripting

SEE RELATED CATEGORY: **Scripting**

- Learn to design and develop powerful scripts.
- Covers AppleScript™, Frontier, QuicKeys, Tempo II, nShell, FaceSpan Application Builder, Scripting PlainTalk and System 7.5.
- Hands on tutorial shows you how to automate your Macintosh activities by learning how to use the AppleScript and Frontier scripting environments.
- Harness the capabilities of a wide variety of Macintosh applications into the integrated productivity tools. This includes such things as the newspaper script which combines the power of SITcomm, MacWrite Pro, and FileMaker Pro, or QuarkXPress.

List $34.95 Our Price **$31.45**

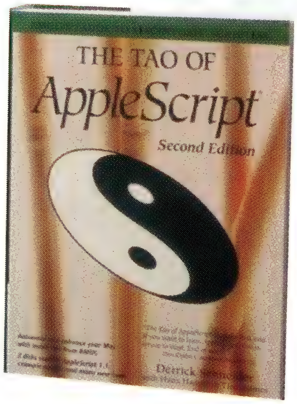
The Tao of AppleScript: BMUG's Guide to Macintosh Scripting, Second Edition by Derrick Schneider & Hans Hansen

- A complete, natural introduction to AppleScript programming essentials.
- Learn to customize applications, automate tedious tasks, and create programs without having to use a complex programming language.
- Includes 2 disks containing AppleScript, QuickTime, StuffIt Lite, ResMover, and other helpful utilities.
- Loaded with practical examples for easy learning

List $29.95 Our Price **$26.95**

SEE RELATED CATEGORY: **Scripting**

Includes Disk!



ScriptBase™ by Main Event Software

- A database for storing persistent objects to be made available for access to AppleScript, Apple's system-level user scripting language for controlling applications on Macintosh® computers.
- Once installed, the database becomes part of the AppleScript system, adding a host of commands to the basic AppleScript vocabulary. Retrieving the objects is simple using AppleScript's natural-language syntax and structure. Objects stored and retrieved in ScriptBase can be accessible any time from any script on the user's computer. These objects can be of any type, including numbers, character strings, lists, records, scripts, and references to disks, files, folders, as well as abstract raw data, to name just a few.
- ScriptBase can be used to maintain system-wide settings, such as sets of preferences, paths to frequently-used files or folders. Complex installations can be made easier by organizing data and scripts within the database's structure.

Our Price **$79.00**

SEE RELATED CATEGORY: **Scripting**

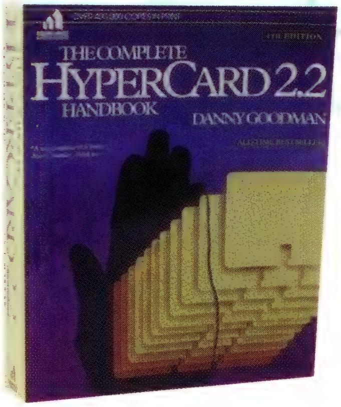
The Complete AppleScript Handbook by Danny Goodman

Master AppleScript with the definitive book/disk toolkit. This self-contained toolkit teaches you AppleScript from the ground up providing you with tools you need to automate tasks and integrate Macintosh applications. This book explains in detail all commands and usage of the AppleScript language including:

• Issuing command • Describing objects • Working with values, variables, and expressions • Using if-then constructions, loops and subroutines • Error checking and debugging • Scripting Finder-level processes • Using AppleScript with third party applications

List $35.00 Our Price **$31.50**

SEE RELATED CATEGORY: **Scripting**



The Complete HyperCard® 2.2 Handbook Fourth Edition

by Danny Goodman

- The biggest-selling programming Mac book.
- Learn to build working applications using the latest version of HyperCard.
- Covers text, painting tools, extension commands (XCMDs), scripting in HyperTalk, and more.

List $35.00 Our Price **$31.50**

SEE RELATED CATEGORY: **Scripting**

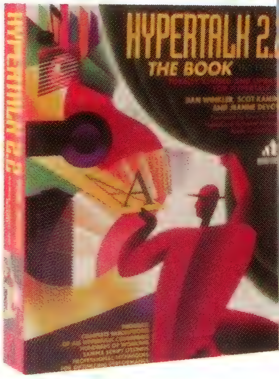
HyperTalk® 2.2: The Book Second Edition

by Dan Winkler, Scott Kamins, and Jeanne DeVoto

SEE RELATED CATEGORY: **Scripting**

- The most complete, authoritative source on HyperTalk 2.2 programming and troubleshooting
- Covers each language element of HyperTalk 2.2 (including the odd quirk or bug).

List $35.00 Ours Price **$31.50**



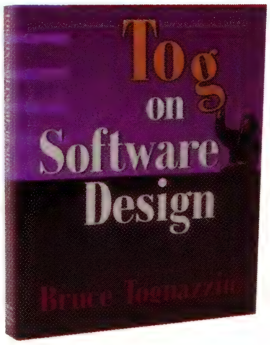
HyperCard Stack Design by Apple Computer, Inc.

HyperCard Stack Design Guidelines is an essential book for everyone who creates Apple® HyperCard stacks. Included are illustrated discussions of:

- Guidelines for stack development-audience evaluation, subject matter requirements and constraints, mode of presentation, and testing
- Navigation, the importance of making sure users can get around in your stacks
- Graphic Design and illustration- including the use of grids to determine card and background layout
- Text in stacks- placement, readability, and special considerations when writing for the screen
- Music and sound in stacks-as subject matter, reinforcement, entertainment, alert mechanism, or transition

List $21.95 Our Price **$19.95**

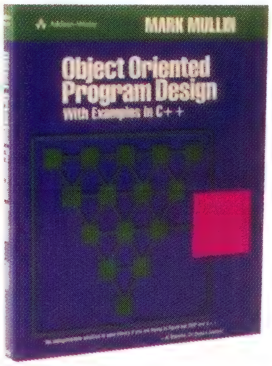
SEE RELATED CATEGORY: **Scripting**



Tog on Software Design by Bruce "Tog" Tognazzini

Respected industry futurist, Tog, presents his vision of our technological future, detailing the steps computer professionals need to take to deliver new technologies that will profit the industry and benefit society in general. Contains Tog's insights on a wide range of topics from quality management to the meaning of standards, and responses to queries supplied by designers and developers.

List $29.95 Our Price **$26.95**



Object Oriented Program Design by Mark Mullin

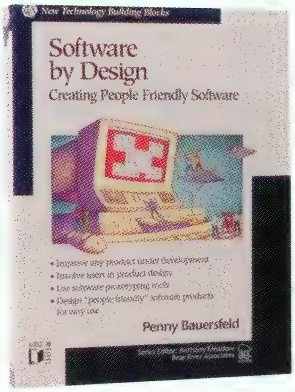
- A concise guide to the essential concepts and techniques of OOP design
- Clarifies the key concepts of object oriented programming such as objects, classes, entities, hierarchies, and inheritance
- Uses typical database application to illustrate each OOP topic, to give the programmer a familiar poi of reference

List $22.95 Our Price **$20.66**

Lowest price guaranteed. Feel Good about your purchase! 30-day return policy.

Software By Design: Creating User Friendly Software

by Penny Bauersfeld (Series Editor: Tony Meadow)


New Technology Building Blocks
Software by Design
Creating People Friendly Software
Penny Bauersfeld

- A thorough how-to for designing software that is easy to learn and comfortable to operate.
- Written from the Macintosh perspective, but compatible with all platforms.
- Stresses user input from initial design, through prototyping, testing and revision.
- Provides tools for analyzing user needs and test responses, plus exercises for sharpening user-oriented design skills.

List $29.95 Our Price **$26.95**

Inside Macintosh®: CD-ROM by Apple Computer, Inc.

- More than 25 volumes in electronic form.
- Includes: QuickDraw™ GX Library, Macintosh Human Interface Guidelines, PowerPC System Software, Macintosh Toolbox Essentials and More Macintosh Toolbox, QuickTime and QuickTime Components.
- Access over 16,000 pages of information with Hypertext linking and extensive cross referencing.

List $99.95 Our Price **$89.95**

Inside Macintosh®: Sound by Apple Computer, Inc.

- Describes the parts of the Macintosh system software that allow you to manage sounds.
- Contains information to write applications that can record and play back sounds, compress and expand audio data, convert text to speech, and perform other similar operations.

List $26.95 Our Price **$24.25**

Inside Macintosh®: Imaging by Apple Computer, Inc.

- Covers QuickDraw and Color QuickDraw.
- Includes general discussions of drawing and working with color.
- Describes the structures that hold images and image information, and the routines that manipulate them.
- Covers the Palette, Color, and Printing Managers, and the Color Picker, Color Matching, and Picture Utilities.

List$26.95 OurPrice **$24.25**

Inside Macintosh®: QuickTime by Apple Computer, Inc.

- For developers who want to create applications that use QuickTime, the system software that allows the integration of video, animation, and sounds into applications.
- Describes all of the QuickTime Toolbox utilities.
- Provides the information you need to compress and decompress images and image sequences.

List $29.95 Our Price **$26.95**

Inside Macintosh®: QuickTime Components by Apple Computer, Inc.

Covers how to use and develop QuickTime components such as image compressors, movie controllers, sequence grabbers, and video digitizers.

List $34.95 Our Price **$31.45**

One & 2-Day SHIPPING
(call for details)

Inside Macintosh®: QuickDraw™ GX Programmer's Overview

- Provides an introduction to QuickDraw™ GX, providing an overview of the QuickDraw GX environment from a developer's perspective.
- Introduces the QuickDraw™ GX programming and runtime environments, the relationship between QuickDraw GX and the rest of the Macintosh® systems software and the relationship between QuickDraw GX and Macintosh applications.
- Learn the key elements of QuickDraw GX programming, data structures, object types, and functions used most frequently by QuickDraw GX developers are also covered.
- Provides a series of practical examples demonstrating how to approach programming with QuickDraw GX.

List $24.95 Our Price **$22.45**

Inside Macintosh®: QuickDraw™ GX Graphics

by Apple Computer, Inc.

- Shows how to create and manipulate the fundamental geometric shapes of QuickDraw GX to generate a vast range of graphic entities.
- Demonstrates how to work with bitmaps and pictures, and specialized QuickDraw GX graphic shapes.

List $26.95 Our Price **$24.25**

Inside Macintosh®: QuickDraw™ GX Objects

by Apple Computer, Inc.

Introduces QuickDraw GX and its object structure, and shows programmers how to manipulate objects in all types of programs.

List $26.95 Our Price **$24.25**

Inside Macintosh®: QuickDraw™ GX Printing by Apple Computer, Inc.

- Essential for any developer whose QuickDraw™ GX application supports printing.
- Shows how to support the new printing features of QuickDraw GX, including desktop printers and expandable printing dialog boxes.
- Shows how to use printing-related objects to add custom panels to printing dialog boxes and to create custom page formats.

List $26.95 Our Price **$24.25**

Inside Macintosh®: QuickDraw™ GX Printing Extensions and Drivers by Apple Computer, Inc.

- Essential for developers who want to create extensions to the application printing capabilities of QuickDraw™ GX, or who need to write a printing device driver that works with QuickDraw GX.
- Describes how to create printing extensions and printer drivers, and provides a complete reference to the messages, functions, and resources that they use.

List $29.95 Our Price **$26.95**

Inside Macintosh®: QuickDraw™ GX Typography by Apple Computer, Inc.

- Essential for developers who use QuickDraw™ GX to manipulate text.
- Shows how to use QuickDraw GX objects to handle all kinds of text – from plain, unstyled text to complex, mixed-direction and multi-language text with sophisticated stylistic and typographic variations.
- Shows how to create and manipulate the three different types of text shapes supported by QuickDraw GX including text shapes, glyph shapes, and layout shapes.

List $29.95 Our Price **$26.95**

Inside Macintosh®: QuickDraw™ GX Environment and Utilities by Apple Computer, Inc.

- Companion to QuickDraw™ GX Objects.
- Contains programming information useful to any developer writing QuickDraw GX applications.
- Describes QuickDraw GX memory management, error handling, debugging, and mathematical functions, as well as conversion from QuickDraw to QuickDraw GX.

List $29.95 Our Price **$26.95**

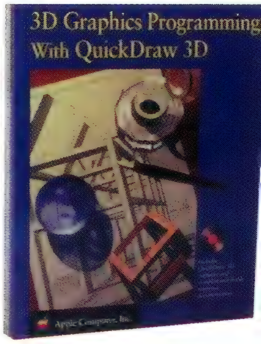
3D Graphics Programming Using QuickDraw 3D

by Apple Computer, Inc.

Includes CD-ROM!

- Incorporate spectacular 3D graphics into your applications.
- Explore QuickDraw 3D, a revolutionary graphics extension to the Mac OS for Power Macintoshes.
- CD contains the complete QuickDraw 3D system itself and a complete database of the QuickDraw 3D API, allowing you instant access to the hundreds of graphics calls via a fast viewing engine. Book/CD-ROM, 640 pages.

List $39.95 Our Price **$35.96**



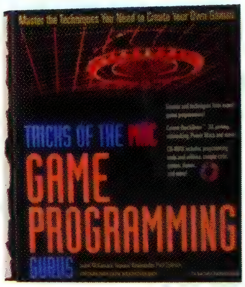
Developer DEPOT

Can't find what you're looking for?
Check our Web site: http://www.devdepot.com

3D Game Machine v1.2 by Virtually Unlimited

- Create lightning-fast 3D arcade games and interactive multimedia applications.
- Ultra-fast rendering – 15 frames per second on a 14" monitor completely texture-mapped, with a PowerMac 6100/60
- Create full "virtual" 3D worlds with six degrees of freedom, free-form texture mapping, shading, material and light properties, convex/cave polygons with unlimited vertices, unlimited light sources, dynamic hidden surface removal, special graphic modes for fast full-screen animation, collision detection, explosion simulation, 3D data importing.
- Simple easy-to-use interface. Runs on all Macs! Works with CodeWarrior.

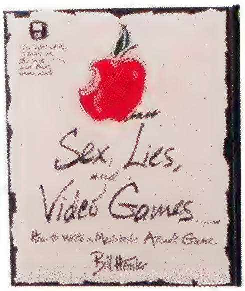
Our Price **$299.00** + license.



Tricks of The Mac Game Programming Gurus

- For beginning to expert game programmers
- Complete overview of all the necessary components of game programming on the Macintosh.
- Packed with valuable tools, utilities, sample code, CodeWarrior™ Lite and game demos.
- QuickDraw 3D and Power Mac optimization and inside info on how Glypha III was created.
- Hundreds of tried-and-true tricks, tips, and insider secrets from well-known Mac game programming experts

List $50.00 Our Price **$45.00**


Sex, Lies, and Video Games
How to write a Macintosh Arcade Game
Bill Hensler

Sex, Lies and Video Games

by Bill Hensler

- A learn-by-example tutorial on the ins and outs of Mac arcade-style game programming in C.
- Features game theory, sprite animation, sound, and interaction techniques.
- A must-read for serious programmer's and hobbyists alike.

List 34.95 Our Price **$31.46**

Includes CD-ROM!

Black Art of Macintosh Game Programming: by; Kevin Tieskoetter.

- Develop your own 3D games in C on the Mac.
- Includes CD with project files for both Symantec C and Code Warrior
- Create freeform texture-mapped games and polygon graphics
- Control dynamic source code-all compatible as native to the Power Mac
- Write directly to the screen, bypassing QuickDraw

List $39.99 Our price **$35.99**

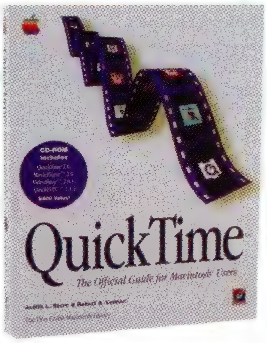
Includes CD-ROM!

QuickTime Official Guide for Mac

- Includes step by step lessons and hands on advice to help you get the most out of QuickTime
- Customize the way you play QuickTime movies
- Digitize your own video
- Cut and Paste clips and add tracks to movies
- With CD that includes QuickTime 2.0, VideoShop 2.0.3, QuickFLIX 1.1.1, and MoviePlayer 2.0
- Incorporate sound, music, and MIDI support in your movies
- Edited by Don Crabb

List $49.50 Our price **$45.00**

SEE RELATED CATEGORY: **Tools, Libs & Utilities**



QuickTime Starter Kit For The Macintosh by Robert A. Lettieri & Judith Stern

- The ultimate package for getting productive and having fun with Macintosh movie-making
- Play, make, and edit QuickTime movies easily
- Loaded with tips on the best ways to bring live-action video to Mac multimedia
- CD-ROM includes QuickTime tools, movie clips, shareware, and demos of Premiere and other programs.

List $45.00 Our Price **$40.50**

SEE RELATED CATEGORY: **Tools, Libs & Utilities**

Graphic Gems V Edited by Alan W. Paeth

- Loaded with practical tools for implementing new ideas and techniques, to offer working solutions to real programming problems.
- Contains over 40 new gems in ellipses, splines, Bezier curves, and ray tracing – displaying the most recent and innovative techniques in graphics programming.
- Includes a disk with source code from all five volumes. Available in both IBM and Macintosh versions. CONTENTS: Algebra and Arithmetic. Computational Geometry. Modeling and Transformation. Curves and Surfaces. Ray Tracing and Radiosity. Halftoning and Image Processing. Utilities.

List $49.95 Our Price **$44.95**

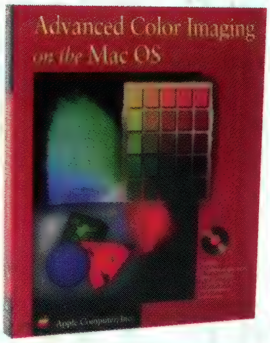
Includes Disk!

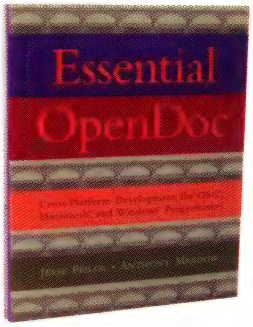
Advanced Color Imaging on the Mac OS

Includes CD-ROM!

- Enhance your software's color capabilities with step-by-step instructions.
- Augment the color support supplied with QuickDraw, and QuickDraw GX.
- Use the Pallette Manager to get the best colors on limited displays.
- Match colors between screens and input/output devices (scanners & printers)
- CD includes a complete reference information in both QuickView and Acrobat formats. Plus, a sample application demonstrating ColorSync programming techniques.

List $36.95 Our Price **$33.25**




Essential OpenDoc
Cross-Platform Development for OS/2, Macintosh, and Windows Programmers

OpenDoc Programmer's Cookbook

- Shows you how to create OpenDoc software components, called parts editors, for the Mac OS Platform.
- Including instructions for setting up the Macintosh Programmers Workshop (MPW) development environment to write OpenDoc software
- Annotated listings of explaining the methods that implement the SamplePart part editor
- Descriptions of other sample part editors created by the OpenDoc engineering team to illustrate more advanced features
- Summary descriptions of software utilities provided with OpenDoc for the Mac OS
- An Introduction to the System Object Model (SOM) technology underlying OpenDoc

List $24.95 Our price **$22.45**

Essential OpenDoc

- Gives an in-depth look at the technical issues of OpenDoc
- Explores the three core technologies that support it's functionality – SOM, OpenDoc's storage mechanism, and the Open Scripting Architecture (OSA).
- Also examines CyberDog, a set of OpenDoc part editors that provides access to Internet services and offers compelling example of the power of OpenDoc development

List $39.95 Our price **$35.95**

Inside CodeWarrior 9

- Includes CodeWarrior IDE User's Guide.
- This is the printed version of the documentation provided on the CD.
- Covers CodeWarrior, the debugger, and associated tools.

Our Price **$34.95**

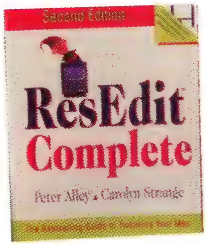
SEE RELATED CATEGORY: **Dev. Environments**

ResEdit™ Complete, Second Edition

by Peter Alley and Carolyn Strange

- Customize every aspect of your interface form creating screen backgrounds and icons to customizing menus and dialog boxes. 608 pages. Book/disk package.

List $34.95 Our Price **$31.45**



Inside PowerPlant Manual

- Create PowerPlant applications using the CodeWarrior IDE and PowerPlant Constructor.
- Full descriptions of major PowerPlant classes and resources.
- Included are the PowerPlant Constructor Manual, including View, TextTraits and Custom Types editing, and PowerPlant Library Reference, covering all classes and functions in PowerPlant.

Our Price **$34.95**

SEE RELATED CATEGORY: **Dev. Environments**

OpenDoc Programmer's Guide by Apple Computer, Inc.

- The official reference for the implementation of OpenDoc on the Mac OS.
- Describes the component software revolution and explains how to develop for it on the Mac OS platform.
- Accompanying CD-ROM contains a complete reference to the OpenDoc programming interface, and an extensive collection of tested, reusable sample code.

List $44.95 Our Price **$40.46**

One & 2-Day SHIPPING (call for details)

C++ Programming with CodeWarrior by Jan L. Harrington

- Beginning OOP for the Macintosh and Power Macintosh and Mac OS compatibles.
- Learn object-oriented programming techniques using C++ as the example language and Metrowerks and CodeWarrior as the example compiler.
- Enclosed CD contains example code from the book and a full-function Metrowerks CodeWarrior

List $35.95 Our Price **$32.35**

Includes CD-ROM!

Developer DEPOT

Check out hundreds of more products on our Web site: http://www.devdepot.com

The ResEdit All Night Diner by David Ciskowski

- An idea-filled menu and introduction to the joys of customizing software.
- Add personality to the Mac by customizing default icons, the text of menus and dialog boxes, cursors, pointers and more.
- Disk features ResEdit, plus lots of sample resources

Includes Disk!

List $24.95 Our Price **$22.45**

SEE RELATED CATEGORY: **Dev. Environments**

C++ Programming W/MacApp by David Wilson, Larry Rosenstein & Dan Shafer

- Learn the secrets to unlocking the power of MacApp®, Apple's development environment for C++
- Learn to design complex windows and views using the ViewEdit tool
- Learn to support multipage text and graphics with only five lines of code
- Learn to support Undo for menu commands and drawing operations that use the mouse.

List $34.95 Our Price **$31.46** SEE RELATED CATEGORY: **Tools, Libs & Utilities**

Programming in Symantec C++ for the Macintosh by Judy May and John Whittle

- An introduction to object-oriented programming, the C++ language, and Symantec C++ for the Macintosh.
- Great for both programmers and beginners alike.
- Covers everything from the basics to advanced features of Symantec C++.
- Includes helpful examples of C++ code that illustrate object-oriented programs.

SEE RELATED CATEGORY: **Dev. Environments**

List $29.95 Our Price **$26.95**

Symantec C++ Programming by Neil Rhodes & Julie McKeehan

Symantec C++ Programming for the Macintosh is a tutorial for getting up and running in the Symantec C++ environment, while mastering the techniques of object-oriented programming.

- Explore the Symantec C++ environment, from debugging a program and using resource utilities to building applications and
  creating objects
- Design programs for compatibility with multiple application frameworks
- Learn how to use the new Visual Architect, Inspector, and templates.

SEE RELATED CATEGORY: **Dev. Environments**

Metrowerks CodeWarrior Programming by Dan Parks Sydow

- Includes CodeWarrior Lite, and Full Coverage of PowerPlant™.
- The best information on Metrowerks CodeWarrior, giving full coverage to the Gold Edition.
- CD includes Code Warrior Lite.

List $39.95 Our Price **$35.95** SEE RELATED CATEGORY: **Dev. Environments**

Includes CD-ROM!

Dan Shafer Presents the Power of Prograph CPX

- Master the revolutionary graphical object-oriented programming language.
- Step by step course through three interrelated projects of increasing complexity.
- Learn Prograph language, CPX classes and object editors.
- Includes disk with all code in the book.

List $49.95 Our Price **$44.95**

Order Toll-free
800-MACDEV-1
(800-622-3381)

Visual Programming with Prograph CPX

by Scott B. Steinman and Kevin G. Carver

- An introduction to the language and a guide for advanced users, for both Macintosh and Windows-based machines.

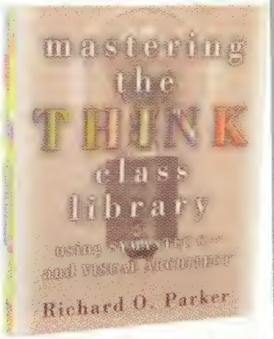
List $34.00 Our Price **$30.60**

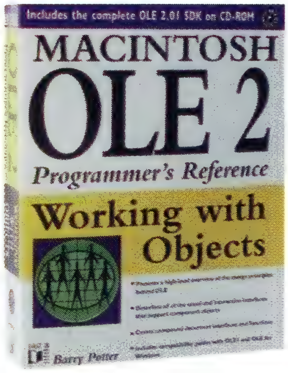
Mastering the THINK Class Library by Richard Parker

- Provides a thorough examination of Symantec's extensive Class Library and the Visual Architect.
- A complete description of the structure and operation of the TCL includes explanations of all code generated by the Visual Architect, any necessary custom code, and the operation of this code.
- Visual Architect tutorials provide you with a step-by-step approach for simplifying the development of complex Macintosh applications. 496 pages.

List $29.95 Our Price **$26.95**

SEE RELATED CATEGORY:
Dev. Environments





Macintosh OLE2 Programmer's Reference: Working with Objects

- Complete reference to the extensible protocol of Object Linking and Embedding, version 2.01 for Macintosh System 7.
- Describes the visual and interactice interfaces that support the component objects.
- Provides details of the OLE 2.01 for the Macintosh user Interface, addresses the issues of object class registration, shows how to implement the drag and drop objects from one application to another, covers the interface that exposes the basic embedding functionality, includes descriptions of API functions and more!

List $44.95 Our Price **$40.45**

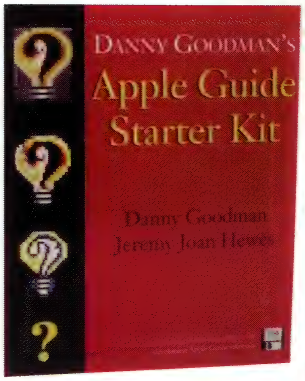
Includes CD-ROM!

Danny Goodman's Apple Guide Starter Kit

by Danny Goodman and Jeremy Joan Hewes.

- Create your own Apple Guide databases quickly and easily, without having to learn a scripting language, write coded files, or use several different files and programs
- Includes advice and tips on how to design a good Guide, from planning and creation through testing, revising, and indexing. Book/disk, 320 pages.

List $34.95 Our Price **$31.46**



MacsBug Reference & Debugging Guide For MacsBug version 6.2

by Apple Computer, Inc.

- MacsBug is an assembly-language-level debugging tool
- Macros, templates, dcmds, and other resources for making debugging easier
- Macintosh memory management and the operating system as they relate to low level debugging
- How to display and set memory and process registers
- Disassemble memory, and set execution breakpoints
- Discipline, a tool for testing the validity of toolbox parameters
- Debugging strategies you can use to find and cure common bugs
- Includes MacsBug 6.2.

List $34.95 Our Price **$31.46**

AppleGuide Complete

by Apple Computer, Inc.

- Covers Guide Maker, the software you use to build and test guide files.
- Learn about the complete cycle of designing as well as advanced topics such as scripting and coding guide files. Book/CD-ROM, 544 pages.

List $39.95 Our Price **$35.96**

Includes CD-ROM!

SEE RELATED CATEGORY:
Tools, Libs & Utilities

Programming For The Newton: Software Development using NewtonScript

by Julie McKeehan and Neil Rhodes. Foreword by Walter R. Smith

- An indispensable tool for Newton programmers.
- Includes disk with sample Newton application from the books, as well as demonstration version of Newton Toolkit (NTK) – the complete development environment for the Newton®.
- A Publication of AP Professional May 1994, Paperback, 393 pp.

List $29.95 Our Price **$26.95**

SEE RELATED CATEGORY:
Dev. Environments

Lowest price guaranteed. Feel Good about your purchase! 30-day return policy.

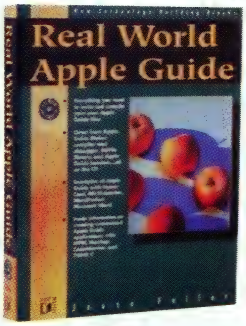
Real World AppleGuide For The Mac

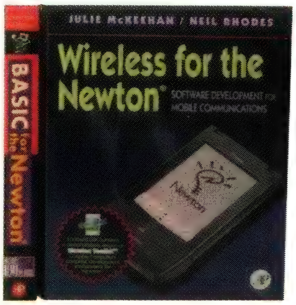
SEE RELATED CATEGORY:
Tools, Libs & Utilities

Includes Disk!

- An introduction to AppleGuide for programmers. It explains its design and function, plus how to design your own guides using AppleScript.
- Includes a disk of sample AppleGuides for AppleGuide-compliant applications.

List $39.95 Our Price **$35.95**





Wireless For The Newton: Software Development for Mobile Communications

by Julie McKeehan and Neil Rhodes

- Learn to develop Newton® software on the Macintosh.
- Hands-on Newton environment training with sample code
- Includes disk with sample source code for a Newton application, as well as demonstration NTK™ – the complete development environment for the Newton®.

List $34.95 Our Price **$31.45**

Programming for the Newton Using NS BASIC

by John Schettino & Liz O'Hara

Includes Disk!

- Program on Macintosh, Windows-based PC, or on the Newton itself.
- Straight-forward "programming by example" approach – you'll be writing Newton programs right away.
- Includes 3.5" disk containing Demonstration NS BASIC and over fifty example programs. (Newton not included)

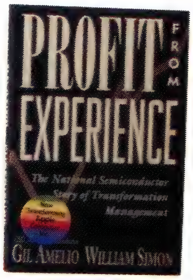
List $35.95 Our Price **$32.35**

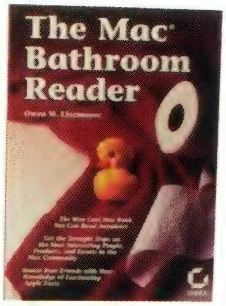
SEE RELATED CATEGORY:
Dev. Environments

Profit From Experience by Gil Amelio and William Simon

Written by the new CEO of Apple

- The story of the transformation of National Semiconductor – how Amelio and his management team took it from it's worst loss in 30 years to the highest earnings in it's history • Includes: 6 core business issues and why they are critical to success • The TEAM program • 10 personal attributes to strive for • Amelio's guidelines for General Managers • Attributes to look for when hiring. List $24.99 Our price **$22.45**


PROFIT FROM EXPERIENCE
The National Semiconductor Story of Transformation Management
GIL AMELIO WILLIAM SIMON



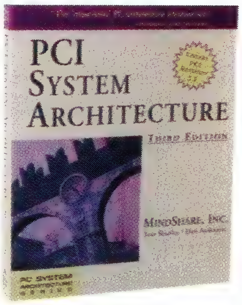
The Mac Bathroom Reader by Owen Linzmayer

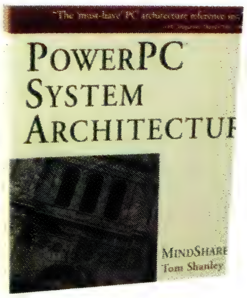
- Get the straight dope on the most interesting people, products, and the events in the Mac community
- Amaze your friends with your knowledge of fascinating Apple facts
- Won't improve your Mac skills, but it will will help you impress your friends with useless Mac trivia.

List $12.99 Our Price **$11.70**

PCI System Architecture, Third Edition by MindShare

Describing revision 2.1 of the Peripheral Component Interconnect (PCI) bus specification, this book explores PCI's relationship to the rest of the system. It includes an in-depth treatment of PCI to PCI bridges, the PCI BIOS, the 66MHz PCI bus, and more. 592 pages. List $34.95 Our Price **$31.46**





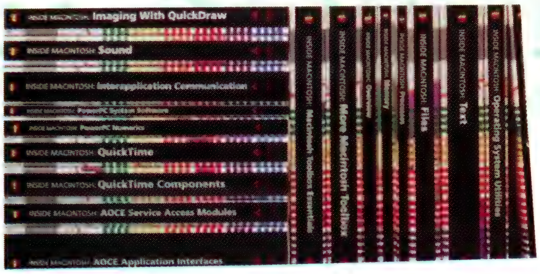
PowerPC System Architecture

by MindShare Inc. and Tom Shanley

- Describes the hardware architecture of the PowerPC systems, providing clear, concise explanation of the PowerPC specification, the template upon which all PowerPC processors are designed.
- Includes the specs for both the 32 and 64 bit implementations including • supervisor privilege level facilities • logical memory addresses • I/O and memory mapped I/O • address translation for segments, pages, and blocks • virtual paging • interrupts.

List $34.95 Our Price **$31.46**

Check out hundreds of more products on our Web site: http://www.devdepot.com


Inside Macintosh Imaging With QuickDraw
Inside Macintosh Sound
Inside Macintosh Interapplication Communication
Inside Macintosh PowerPC System Software
Inside Macintosh PowerPC Numerics
Inside Macintosh QuickTime
Inside Macintosh QuickTime Components
Inside Macintosh AOCE Service Access Modules
Inside Macintosh AOCE Application Interfaces
Inside Macintosh Macintosh Toolbox Essentials
Inside Macintosh More Macintosh Toolbox
Inside Macintosh Overview
Inside Macintosh Memory
Inside Macintosh Processes
Inside Macintosh Files
Inside Macintosh Text
Inside Macintosh Operating System Utilities

Inside Macintosh®: AOCE Application Interfaces

by Apple Computer, Inc.

- Shows how your application can take advantage of the system software features provided by PowerTalk system software and the PowerShare collaboration servers.
- Add electronic mail capabilities to your application, write a messaging application or agent, store information in and retrieve information from PowerShare and other AOCE catalogs.
- Add catalog-browsing and find-in-catalog capabilities to your application, write templates that extend the Finder's ability to display information in PowerShare and other AOCE catalogs.
- Add digital signatures to files or to any portion of a document, and establish an authenticated messaging connection.

List $40.45 Our Price **$36.40**

Inside Macintosh®: Interapplication Communication

by Apple Computer, Inc.

Shows how applications can work together. How your application can share data, request information or services, allow the user to automate tasks, communicate with remote databases.

List $34.95 Our Price **$31.45**

Inside the Macintosh Communications Toolbox

by Apple Computer, Inc.

- The definitive reference to the Macintosh Communications Toolbox, an integral part of the System 7 Macintosh Toolbox that enables developers to create communications applications or add communications features to other applications.
- Describes all of the routines that provide programmers with standard access to important communications services and in addition enables programmers to extend the reach of the Macintosh into non-Apple environments.

List $24.95 Our Price **$22.95**

Inside Macintosh®: Networking

by Apple Computer, Inc.

- Describes how to write software that uses AppleTalk networking protocols
- Describes the components and organization of AppleTalk and how to select an AppleTalk protocol.
- Provides the complete application interfaces to all AppleTalk protocols, including ATP (AppleTalk Transaction Protocol), DDP (Datagram Delivery Protocol), and ADSP (AppleTalk Data Stream Protocol), among others.

List $29.95 Our Price **$26.95**

Inside Macintosh®: AOCE Service Access Modules **by Apple Computer, Inc.**

- Describes how to write a software module that gives users and PowerTalk-enabled applications access to a new or existing mail and messaging service or catalog service.
- Shows how to write a catalog service access module (CSAM), a messaging service access module (MSAM), and AOCE templates that allow a user to set up a CSAM or MSAM and add addresses to mail and messages.

List $26.95 Our Price **$24.25**

Inside Macintosh®: Text **by Apple Computer, Inc.**

- Describes how to perform text handling, from simple character display to multi-language processing.
- Covers Font, Script, Text Services, and Dictionary Managers, in addition to QuickDraw Text, TextEdit, and International and Keyboard Resources.

List $39.95 Our Price **$35.95**

Inside Macintosh®: Devices **by Apple Computer, Inc.**

- Describes how to write software that interacts with built-in and peripheral hardware devices.
- Learn how to write and install your own device drivers, desk accessories, and Chooser extensions.
- Communicate with device drivers using the Device Manager; access expansion cards using the Slot Manager; control SCSI devices using SCSI Manager 4.3 or the original SCSI Manager.
- Communicate directly with Apple Desktop Bus devices; interact with the Power Manager in battery-powered Macintosh computers, and communicate with serial devices using the Serial Driver.

List $29.95 Our Price **$26.95**

Inside Macintosh®: X-Ref **by Apple Computer, Inc.**

Is a fast access to all the information in Inside Macintosh. Inside Macintosh X Ref; Provides programmers with a quick and easy way to find the exact information they need in this definitive suite of books, (all 26 volumes). It is indexed by topic, volume, chapter, and accompanying page number.

List $19.95 Our Price **$17.95**

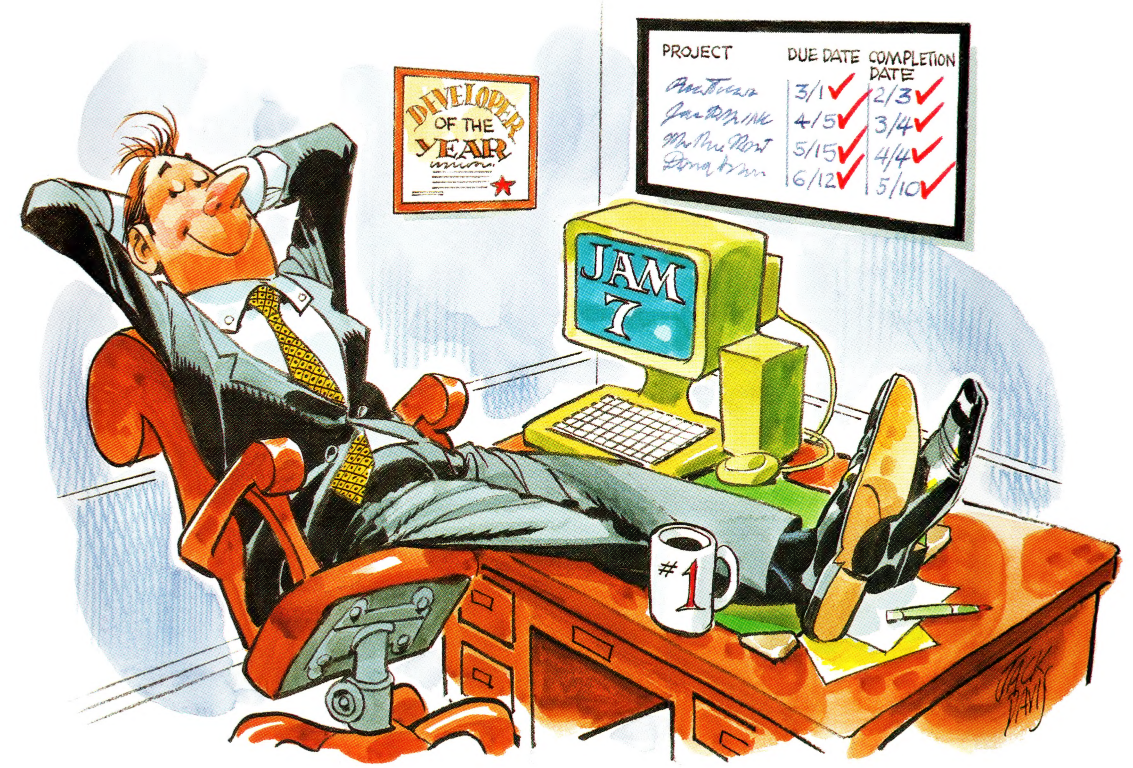
Order Toll-free
800-MACDEV-1
(800-622-3381)

All entries in this index are alphabetized.
For your convenience the product names are **bold**, book names are *italicized* and company names are in plain text.

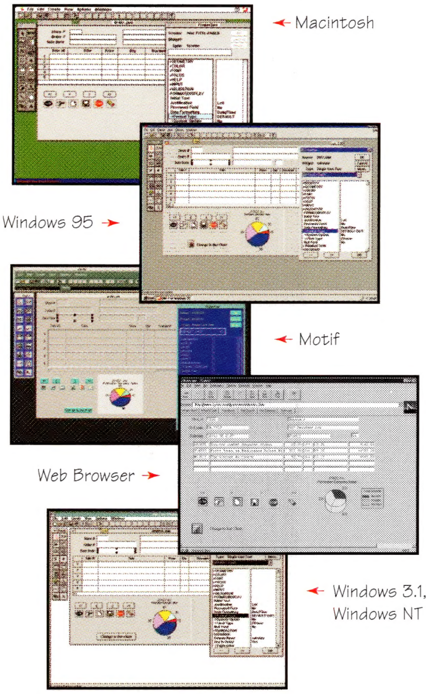
3D Game Machine 25
3D Graphics Programming Using QuickDraw 3D 24
Absoft 5
Adianta 10
Advanced Color Imaging on the Mac OS 26
Amplified Intelligence 12
AppleGuide Complete 28
AppleScript Finder Guide 20
AppleScript Language Guide 20
AppleScript Scripting Additions Guide 20
Applied Mac Scripting 21
AppMaker 14
B-Tree Helper 13
Bare Bones Software 8
BASIC for the Newton 5
BBEdit 8
Bowers Software 14
Black Art of Macintosh Game Programming 25
C++ Programming w/MacApp 27
C++ Programming with CodeWarrior 26
CLImate 9
CMaster 8
CodeManager 9
CodeWarrior 3
Complete AppleScript Handbook 21
Complete HyperCard 2.2 Handbook 22
Computer Privacy Handbook 16
CPU Doubler 9
CronManager 10
Cyberpunk Handbook, The Real Cyberpunk Fakebook 16
Dan Shafer Presents the Power of Prograph CPX 27
Danny Goodman's AppleGuide Starter Kit 28
Danny Goodman's AppleScript Handbook 21
DataScript 7
Discover Programming for Macintosh 17
dtF 3
Duet Development 9
E-Mail Essentials 16
Elements of E-Mail Style 16
Emerson Kennedy 11
Essential OpenDoc 26
Excel Software 14
FaceSpan 7
File Genie Pro 9
Fortner Research 4, 5
Fortran 77SDK 5
Fragment of Your Imagination 19
FrameWorks Magazine/Disks 2
Graphic Gems V 25
Guide Composer 10
Hooked on Java 16
HyperCard Stack Design 22
HyperTalk 2.2: The Book 22
ICONIX Power Tools 14
InCider 11
Inside CodeWarrior 8 26
Inside Macintosh: AOCE Application Interfaces 30
Inside Macintosh: AOCE Service Access Modules 30
Inside Macintosh: Devices 30
Inside Macintosh: Files 15
Inside Macintosh: Imaging 23
Inside Macintosh: Interapplication Communication 30
Inside Macintosh: Macintosh Toolbox Essentials 15
Inside Macintosh: Memory 15
Inside Macintosh: More Macintosh Toolbox 15
Inside Macintosh: Networking 30
Inside Macintosh: Operating System Utilities 15
Inside Macintosh: Overview 15
Inside Macintosh: PowerPC Numerics 20
Inside Macintosh: PowerPC System Software 20
Inside Macintosh: Processes 15
Inside Macintosh: QuickDraw GX Environment and Utilities 24
Inside Macintosh: QuickDraw GX Graphics 24
Inside Macintosh: QuickDraw GX Objects 24
Inside Macintosh: QuickDraw GX Printing 24
Inside Macintosh: QuickDraw GX Printing Extensions and Drivers 24
Inside Macintosh: QuickDraw GX Programmer's Overview 23
Inside Macintosh: QuickDraw GX Typography ... 24
Inside Macintosh: QuickTime 23
Inside Macintosh: QuickTime Components 23
Inside Macintosh: Text 30
Inside Macintosh: X-Ref 30
Inside PowerPlant Manual 26
Inside the Macintosh Communications Toolbox 30
Instant Internet Guide 16
Java in a Nutshell 16
JavaScript for the Macintosh 16
Jersey Scientific 8
Last Resort Programmers Edition 11
Late Night Software 7
Learn C on the Macintosh 17

Learn C++ on the Macintosh 18
LJ Profiler 11
LPA MacProlog 5
LS Fortran 5
LS Object Pascal CD-ROM 4
Mac Bathroom Reader 29
MacA&D 14
MacAnalyst/Expert 14
MacDesigner/Expert 14
MacFortran II 5
MachTen Power Unix 4
Macintosh C Programming Primer Volume 1 18
Macintosh C Programming Primer Volume 2 18
Macintosh OLE2 Programmer's Reference: Working with Objects 28
Macintosh Pascal Programming Primer Volume I 18
Macintosh Programming Secrets 18
Macintosh Programming Techniques 19
MacsBug Reference & Debugging Guide 28
MacTech CD-ROM 2
MacTech Magazine 2
MacTech Mouse Pad 2
MacTutor, Best of 2
MacWireFrame 12
MADACON '93 CD-ROM 2
Mastering the THINK Class Library 28
Memory Mine 10
Metrowerks 3, 9
Metrowerks CodeWarrior Programming 27
Mjølner BETA System 5
More Macintosh Programming Techniques 19
Movie Cleaner Pro 8
Natural Intelligence, Inc. 6
NeoAccess 13
NeoLogic 13
Nisus Software 8
Object Oriented Program Design 22
Onyx Technology 11
OOFile 13
OOFILE HTML Writer 6
OpenDoc Programmer's Cookbook 26
OpenDoc Programmer's Guide 26
Optimizing PowerPC Code: Programming the PowerPC in Assembly Language 20
Orchard Software 9, 10
Paradigm Software 12
PCI System Architecture 29
Personal MacTen 4
Picture CDEF 12
Planning and Managing Websites 16
Power Macintosh Programming Starter Kit 18
PowerPC System Architecture 29
PowerTap 11
PreFab Player 7
PreFab Software, Inc. 7
Presenting Magic Cap 3
Profit from Experience 29
Programmer's Toolbox Assistant CD-ROM 15
Programming for the Newton using NS BASIC 29
Programming for the Newton: Software Development using NewtonScript 28
Programming in Symantec C++ for the Macintosh 27
Programming Primer Macintosh Volume 1 17
Programming QuickDraw 19
Programming with AppleTalk 19
Q3S/3dPane/SmartPane 13
QC 11
Quasar Knowledge Systems 3
QUED/M 8
QuickTime Official Guide 4 Mac 25
QuickTime Starter Kit for the Macintosh 25
Real World AppleGuide for the Mac 29
ResEdit All Night Diner 27
ResEdit Complete 26
Roaster 6
Rosanne 10
Script Debugger 7
ScriptBase 21
Scripter 7
ScriptGen Pro 10
ScriptWizard 7
Sex, Lies and Video Games 25
SmalltalkAgents 3
SoftPolish CD-ROM 11
Software by Design: Creating User Friendly Software 23
SpellsWell 12
Spyer 11
Step-Up Installer Pack 10
StepUp Software 10
Stone Tablet 12
StoneTable 12
Symantec C++ for 68k 4
Symantec C++ for Power Macintosh 4
Symantec C++ Programming 27
Symantec Corporation 4
Tao of AppleScript: BMUG's Guide to Macintosh Scripting 21
TCP/IP Scripting Addition 6
Teach Yourself Java for Macintosh in 21 Days 16
Tenon Intersystems 4, 10
Tenon Ported Application CD 10
Terran Interactive 8
THINK Pascal 4
Tog on Software Design 22
Tricks of the Mac Game Programming Gurus 25
Visual Programming with Prograph CPX 27
Vivistar Consulting 13
VOODOO 11
Web Weaving 17
WebHead Mac Guide 17
Webmaster Macintosh 17
Wireless for the Newton: Software Development for Mobile Communications 29
Xplain Corporation 2, 5

Life is Sweet with JAM!



Get JAM and enjoy success in building and deploying client/server and Web applications



In a world where requirements change and deadlines don't, only JAM gives you what you need to build powerful, completely portable client/server, 3-tier and Web applications. JAM's cross-platform RAD capabilities enable you to successfully develop and deploy demanding applications across any platform, database, GUI or Web Browser.

Are you willing to base your project's success on a tool that only *promises* portability? JAM lets you build great looking applications that are fully portable *today*. If you need to build serious client/server, 3-tier or Web applications, call JYACC for a free demonstration kit or white paper.

- ✓ Desktop ease with the power of a 2nd-generation tool
- ✓ Unrivaled portability between Windows® 3.1, NT, 95; Macintosh®; OS/2 Warp®; Character, Motif and Web Browsers
- ✓ Unparalleled database access to Oracle®, Sybase®, Informix®, ODBC and more
- ✓ Automatic SQL generation with full transaction control
- ✓ Visual repository-driven development of both client and application server
- ✓ Centralized control over application objects via inheritance
- ✓ Unique architecture for team development
- ✓ No runtimes

Find out for yourself how sweet life can be with JAM!

Call 1 800 458-3313.

Or E-mail: sweetlife3@jyacc.com for a free demonstration kit or white paper.
Visit our Web site at http://www.jyacc.com
For international inquiries call: 1-212-267-7722 or FAX 1-212-608-6753

JYACC
Meeting the needs of professional developers for over 18 years

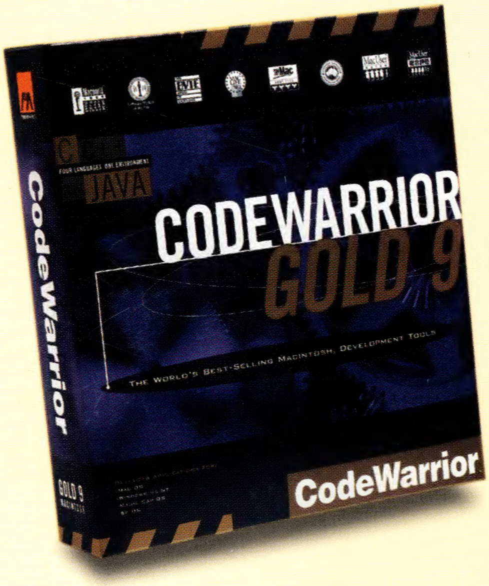
BRAZIL (55) 11 816 6228 • FINLAND (358) 3486 4000 • FRANCE (33) 1 46 92 45 44 • GERMANY (49) 40 79 70 07 0 • HRVATSKA (385) 1 242 116 • ISRAEL (972) 3 557 3675
ITALY (39) 2 45 2701 • MEXICO (52) 5 663 0405 • RUSSIA (7) 095 288 1924 • SAUDI ARABIA (966) 2 665 8406 • SINGAPORE (65) 220 8322 • SLOVENIA (386) 61 1405 004
SPAIN (34) 1 804 0625 • SWEDEN (46) 8 15 10 10 • SWITZERLAND (41) 21 991 9041 • THAILAND (66) 2 513 3559 • THE NETHERLANDS (31) 70 320 9214 • UNITED KINGDOM (44) 171 814 6660

JAM is a registered trademark of JYACC, Inc. Other trademarks are the property of their respective owners.

METROWERKS WORLDWIDE
http://www.metrowerks.com

The future is definitely not running away from you.
Java™ tools for CodeWarrior 9 are here.

CodeWarrior, the award-winning Mac-hosted development suite now includes full-featured Java™ tools! CodeWarrior 9 introduces the Java plugin compiler, linker, preference plugins, source-level debugging, and Java class browsing! All this and more, completely integrated into a Java development environment that you could expect only from CodeWarrior.



CodeWarrior®

CodeWarrior Gold 9 - $399

The choice for multiple OS development from the Mac™ OS. Includes C / C++, Object Pascal and **new Java compiler plugins** for multiple targets. Compiles for 68K Macintosh®, Power Macintosh™, Windows® 95 and Windows NT™ for x86, Magic Cap™and the Be™ OS.

Discover Programming™ with Java - $99

Learn how to program using Java. Designed for both novice and experienced programmers, Discover Programming with Java includes the most complete toolset available on the Macintosh, an electronic version of *Learn Java on the Macintosh* by Barry Boone with Dave Mark and 6 Apple® Guide files, the electronic tutorials that take you step-by-step through the learning process.

Macworld 1996 World Class Award Winner
1994 MacUser Editors' Choice Award Winner
SOFTWARE PRODUCT OF THE YEAR
1994 BYTE Award of Excellence
Mac Life Grand Prix
SVM Mac Août 1995
MacUser Rated 4½
MacUser Japan 総合評価 4½

For More Information Call 1-800-377-5416

System requirements: Macintosh 68020, 68030 or 68040, or Power Macintosh 601, 603 or 604 processor. Minimum of 8-16MB RAM, CD-ROM drive to install software. ©1996 Metrowerks Corporation. All rights reserved. All products are trademarks of their respective companies.

metrowerks®